AD-A203 406

DOCUMENTATION PAGE

Form Approved
OMB No 0704-0188
Exp Date Jun 30, 1986

Unclassified	1b. RESTRICTIVE MARKINGS
2a. SECURITY CLASSIFICATION AUTHORITY	3. DISTRIBUTION/AVAILABILITY OF REPORT: as it appears on the report
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE	
4. PERFORMING ORGANIZATION REPORT NUMBER(S)	5. MONITORING ORGANIZATION REPORT NUMBER(S)
6a. NAME OF PERFORMING ORGANIZATION: HQDA, MILPERCEN; 6b. OFFICE SYMBOL (If applicable): DAPC-OPA-E	7a. NAME OF MONITORING ORGANIZATION
6c. ADDRESS (City, State, and ZIP Code): 200 Stovall Street, Alexandria, VA 22332	7b. ADDRESS (City, State, and ZIP Code)
8a. NAME OF FUNDING/SPONSORING ORGANIZATION: HQDA, MILPERCEN; 8b. OFFICE SYMBOL (If applicable): DAPC-OPA-E	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER
8c. ADDRESS (City, State, and ZIP Code): 200 Stovall Street, Alexandria, VA 22332	10. SOURCE OF FUNDING NUMBERS: PROGRAM ELEMENT NO. / PROJECT NO / TASK NO / WORK UNIT ACCESSION NO

11. TITLE (Include Security Classification): Cross-Coordinated Control: An Experimentally Verified Technique for the Hybrid Twist and Wrench Control of a Voltage-Controlled Industrial Robot

12. PERSONAL AUTHOR(S): Swinson, Mark L.

13a. TYPE OF REPORT	13b TIME COVERED	14. DATE OF REPORT (Year, Month, Day)	15. PAGE COUNT
Final	FROM Aug 85 TO Dec 88	December 30, 1988	274

16. SUPPLEMENTARY NOTATION

17. COSATI CODES			18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
FIELD	GROUP	SUB-GROUP	

19. ABSTRACT (Continue on reverse if necessary and identify by block number)

Applications for modern industrial robots have generally been limited to low-precision, non-contact tasks, because in practice, it is generally not possible to adequately engineer even a manafacturing environment to a sufficiently high degree of precision such that position control, alone, is adequate. This limitation has severely restricted the range of economically justified industrial robot installations. Cross-coordinated control serves to extend that range by providing a practical, experimentally verified solution to the problem of simultaneously controlling both the motion of and the constraint forces acting upon a robot end-effector, which is in contact with a rigid environment. (Continued on reverse.)

20 DISTRIBUTION/AVAILABILITY OF ABSTRACT	21 ABSTRACT SECURITY CLASSIFICATION
☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT ☐ DTIC USERS	Unclassified

22a NAME OF RESPONSIBLE INDIVIDUAL	22b TELEPHONE (Include Area Code)	22c OFFICE SYMBOL
Major Mark L. Swinson	(904) 377-0854	

DD FORM 1473, 84 MAR — 83 APR edition may be used until exhausted. All other editions are obsolete.

SECURITY CLASSIFICATION OF THIS PAGE

19. (Continued)

More precisely, this work provides a semi-empirical method for the hybrid control of a voltage-controlled industrial robot, such that the geometric constraints are explicitly accounted for. The kinestatic analysis was based on a model of the environment using Ball's reciprocal screws to characterize the nature of the constraints. This approach was chosen so as to ensure that the constraint formulation process was invariant with respect to a change of origin, a change of scale or a change of basis.

The resulting theoretical development, combined with a laboratory implementation which employed an instrumented, anisotropic, mechanically complaint wrench sensor, resulted in a system that is kinematically, dynamically and kinestatically stable. Consequently, this approach constitutes a general solution to the problem of performing the commonly encountered industrial tasks which, if automated, would require contact between the robot's end-effector and a rigid environment. Several representative tasks were demonstrated in the laboratory, using a modified, General Electric P60 industrial robot.

The potential for cross-coordinated control to extend the range of economical applications is especially significant, since this method is well suited for implementation as an augmentation, thus permitting the continued use of most existing motion control hardware and software.

Cross-Coordinated Control:
An Experimentally Verified Technique for the Hybrid Twist and Wrench Control of a Voltage-Controlled Industrial Robot

Major Mark L. Swinson
HQDA, MILPERCEN (DAPC-OPA-E)
200 Stovall Street
Alexandria, VA 22332

Final Report - December, 1988

Distribution A

A thesis submitted to the University of Florida in partial fulfillment of the requirements for the degree of Doctor of Philosophy.

CROSS-COORDINATED CONTROL: AN EXPERIMENTALLY VERIFIED TECHNIQUE FOR THE HYBRID TWIST AND WRENCH CONTROL OF A VOLTAGE-CONTROLLED INDUSTRIAL ROBOT

By

MARK L. SWINSON, P.E.

A DISSERTATION PRESENTED TO THE GRADUATE SCHOOL
OF THE UNIVERSITY OF FLORIDA IN
PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

UNIVERSITY OF FLORIDA

1988

TO PRISCILLA, ERIKA AND KARL

I've agreed to this. I'm strapped in. I'm in the hands of something bigger and more powerful than myself. I guess I'll sit back and enjoy the ride.

Jack Nicholson's advice from "Terms of Endearment"

ACKNOWLEDGEMENTS

The author would like to express his sincere appreciation to his advisor, Dr. Joseph Duffy, for his encouragement, guidance and generosity, both in time and material resources. The author would also like to thank the other members of his commitee for their time, efforts and thoughtful suggestions. These include Dr. John Staudhammer, Dr. Ralph Selfridge, Dr. Roy Harrell and Dr. Carl Crane. Also, the author thanks Professor John Rees Jones, Liverpool Polytechnic Institute, Professor Kenneth Hunt, Monash University, Melbourne, Australia, and Professor Emeritus F. Erskine Crossley, University of Massachusetts, for their thoughtful advice, comments, wisdom and wit.

Furthermore, the author would like to thank both his fellow graduate students and the many staff members who freely offered much help and advice. Particularly noteworthy were Mike Griffis, for his thoughtful insight into screw kinetics, Lotfi Romdhane, for his insight into robotic grasping, and Vann Chesney, for his tireless efforts in the laboratory to protect the author from electrocuting himself, or worse, damaging the equipment.

Finally, the author would like to thank his family for their love and support, without which this project could not have been completed. These include his parents, Raymond and Lois, for giving him life and direction, his loving wife Priscilla, for giving him many directions, and his children, Erika and Karl, who are his hope for the future.

This research was funded by the United States Army, the United States Department of Energy and the University of Florida's Center for Intelligent Machines and Robotics.

TABLE OF CONTENTS

Abstract of Dissertation Presented to the Graduate School
of the University of Florida in Partial Fulfillment of the
Requirements for the Degree of Doctor of Philosophy

CROSS-COORDINATED CONTROL: AN EXPERIMENTALLY VERIFIED TECHNIQUE FOR THE HYBRID TWIST AND WRENCH CONTROL OF A VOLTAGE-CONTROLLED INDUSTRIAL ROBOT

By

Mark L. Swinson, P.E.

December 1988

Chairman: Dr. Joseph Duffy
Major Department: Mechanical Engineering

Applications for modern industrial robots have generally been limited to low-precision, non-contact tasks, because in practice, it is generally not possible to adequately engineer even a manufacturing environment to a sufficiently high degree of precision such that position control, alone, is adequate. This limitation has severely restricted the range of economically justified industrial robot installations. Cross-coordinated control serves to extend that range by providing a practical, experimentally verified solution to the problem of simultaneously controlling both the motion of and the constraint forces acting upon a robot end-effector, which is in contact with a rigid environment.

More precisely, this work provides a semi-empirical method for the hybrid control of a voltage-controlled industrial robot, such that the geometric constraints are explicitly accounted for. The kinestatic analysis was based on a model of the

environment using Ball's reciprocal screws to characterize the nature of the constraints. This approach was chosen so as to ensure that the constraint formulation process was invariant with respect to a change of origin, a change of scale or a change of basis.

The resulting theoretical development, combined with a laboratory implementation which employed an instrumented, anisotropic, mechanically compliant wrench sensor, resulted in a system that is kinemetically, dynamically and kinestatically stable. Consequently, this approach constitutes a general solution to the problem of performing the commonly encountered industrial tasks which, if automated, would require contact between the robot's end-effector and a rigid environment. Several representative tasks were demonstrated in the laboratory, using a modified, General Electric P60 industrial robot.

The potential for cross-coordinated control to extend the range of economical applications is especially significant, since this method is well suited for implementation as an augmentation, thus permitting the continued use of most existing motion control hardware and software.

CHAPTER 1
INTRODUCTION

1.1 Background

While certainly being well-suited for repetitive, tedious tasks, robot manipulators were born out of the necessity for machines to be able to perform tasks which are inherently dangerous for human workers. This initially manifested itself as the master-slave manipulators used for radioactive laboratory work in the 1940's [Goertz, 1963]. This first application showed that force feedback was critical if the operator was to be able to perform precise tasks without damaging either the slave manipulator or the workpiece.

Much improvement has been made since those early days. Actuators have replaced human articulation. Sensors, albeit to a very limited extent, have been incorporated. Numerous exotic control strategies have been suggested to improve performance. A few of these control strategies have been simulated, and fewer still have been demonstrated in the laboratory, let alone the factory floor. The end result has been that for all intents and purposes, the industrial robot of today, especially the voltage-controlled robot, remains essentially a pick-and-place device, although certainly much refined as compared with those early manipulators.

The introduction of task-space sensors for robot control implies a level of technology in which the robot can interact intelligently with its environment. As already observed, however, this level of technological sophistication is not characteristic

of modern industrial robots. Rather, the vast majority of industrial robots are used for spray painting, spot welding, or pick-and-place material handling operations where the overall control structure is open-loop. Little, and frequently no feedback concerning the quality, progress or completion of the task is utilized. Clearly, this weakness prevents industrial robots from operating in anything but the most highly structured environments, performing relatively simple, non-contact tasks.

The integration of task-space sensors in closed-loop control offers the possibility of overcoming some of these limitations. The desirability of this is generally recognized, as are many of the associated difficulties. For instance, would such a controller be cost effective? Would the resulting system even be stable? What variables should be sensed, and with what types of sensors?

Generally, industrial robots can respond to sensory feedback involving position, or its time derivatives, and forces. The variety of sensors available is extensive, including acoustic, optoelectronic, vision systems, strain gauges, encoders, resolvers and potentiometers to name but a few [Chesmond, 1982, and Ruocco, 1987]. Such sensors can serve to monitor both task progress and the state of the environment.

No doubt vision, more than any other sensory feedback, has captured the imagination of researchers in this area, and much has been accomplished. However, from a near term perspective, force feedback seems to offer an even higher performance enhancement to cost ratio, given the many immediate applications, including both assembly and contact process tasks.

Nonetheless, control of a tangibly constrained robot end-effector, or "robot force control" as it is usually referred to in the literature [Whitney, 1987], remains well behind vision in both sophistication of theory and in its level of industrial implementation.

Those few implementations tend to fall into one of three broad classes. The first is those tasks for which some nominal trajectory must be executed, while maintaining some specified wrench against the environment. Industrial process applications such as contour following, grinding and deburring are representative of such tasks. The second consists of those tasks for which force sensing provides very high resolution, relative positioning information. Such information is critical in close tolerance assembly. Finally, there is damage prevention, which is particularly suited to force feedback. Even in a stable contact situation the possibility of damage to product or equipment exists if contact forces exceed certain limits.

This research focuses not only on the problems of sensing in one coordinate space and actuating in another, but also on the problems occurring when the variables sensed are of mixed type, namely, forces and motions. Previous attempts to solve this problem have sometimes introduced kinematic instability [An and Hollerbach, 1987b] or kinestatic instability [Lipkin and Duffy, 1986], in addition to the dynamic instability problems usually associated with remote sensing of position or force alone [Eppinger and Seering, 1986]. Hence, the hybrid problem is the more general, with only motion feedback or only force feedback, whether sensors are colocated or not, representing special, simpler cases.

Following a review of previous work in the area of robot force control, as well as a listing of the principal contributions of this research, subsequent chapters detail both the theoretical development and the experimental implementation of cross-coordinated control. A brief description of each chapter is as follows:

Chapter 2: Fundamental Concepts. This chapter introduces the fundamental concepts necessary to understand the essential features of the hybrid control problem,

beginning with a brief introduction to line geometry and screw theory. Additionally, the subject of "kinestatics" is introduced. These ideas are essential to understanding some of the stability problems associated with this class of robot control problems. Other factors related to stability are also discussed. Finally, an introduction to pertinent dc motor characteristics is presented.

Chapter 3: Analysis of Commercial Robot System. This chapter presents an analysis of the General Electric (GE) P60 robot from the standpoint of implementing hybrid force/motion control with a digital computer, with a special emphasis on empirical techniques. Consequently, a somewhat unconventional approach to both the kinematic analysis, and especially the dynamic analysis, is suggested. Also examined is the commercial motion control system. Supporting experimental results are included.

Chapter 4: Conceptual Design. This chapter begins with a description of the control system architecture. The compensator design procedure, which makes use of the results of the system analysis from Chapter 3, is then presented, including a specific example. A formalism for constraint formulation, necessary for the higher level task planning and programming, is then developed. Finally, examples of industrial implementations are presented.

Chapter 5: System Development. This chapter discusses the design and construction of an instrumented, anisotropic, mechanically compliant wrench sensor, as well as the sensor interfacing. Digital computer control issues, such as I/O handling, sampling rates and software are addressed. Finally, the problems of system integration and testing are detailed, which yielded some surprising results.

Chapter 6: Experimental Results. This chapter begins with a presentation of performance results, including both time domain and frequency domain performance

parameters. An analysis of experimental uncertainty is also included. Finally, the laboratory implementation of the industrial applications described in chapter 4 are presented. These applications are representative of many of the contact process and assembly tasks commonly found in a manufacturing environment.

1.2 Review of Previous Efforts

Robot force control has been the subject of a great deal of interest in recent years. This work includes selective joint torque control [Inoue, 1971], damping control [Whitney, 1977], stiffness control [Salisbury, 1980], impedance control [Hogan, 1985, and Kazerooni et al., 1986] and hybrid control [Mason, 1981, and Raibert and Craig, 1981] to name just a few.

It is generally agreed that in order for industrial manipulators to achieve more of their potential, multi-axis force feedback must be effectively integrated into the overall control strategy. Yet despite the large number of papers published in this area, very little experimental work has been reported in the literature on system performance. Quoting from a recent book on robot control, which closely parallels the author's own observations and thoughts on the subject, one finds that

> despite the voluminous publications on the theory of robot control, ranging from PD to nonlinear control, there are almost no experimental results on performance. To be sure, complicated proofs are often given, and occasionally simulations, that supposedly validate an approach. If robot control is to become a scientific endeavor rather than just the pursuit of esoteric mathematics, it must incorporate experimentation to form a critical hypothesize-and-test loop. There is simply no other way to verify convincingly that particular control algorithms work or make a difference, or to guarantee that one is confronting real problems. [An et al., 1988, page 2.]

One early approach to robot force control was suggested by Inoue [1971], and later improved upon by Shimano [1978]. As mentioned previously, it is called selective

joint torque control, and essentially consists of segregating joint axes into those to be position-controlled and those to be force-controlled. Unfortunately, this approach can only be successfully implemented for very simple task geometries, where there is a one-to-one correlation between task and joint axes and, where appropriate, redundant degrees of freedom are available. Hence, this technique generally requires a fairly gross approximation. It does, however, highlight a fundamental conceptual problem of robot force control. Given that it is required to force control specific task axes, while simultaneously controlling the others for position (and velocity), how must the problem be formulated such that the error correction signals sent to the actuators are compatible?

In 1976 it seemed that Mason [1976 and 1979] had begun to provide a satisfactory answer to this question. He introduced a novel, seemingly promising concept of augmenting environmentally imposed "natural" constraints by a set of complementary "artificial" constraints. This work was aimed at formalizing the high-level methodology, and suggested the formation of filters which could modify the specified twists and wrenches to make them consistent with the environment.

Modern hybrid control theory, which stemmed from this intuitive theoretical work of Mason, became the subject of growing research effort. There was apparent experimental verification by Craig and Raibert [1979], and by Raibert and Craig [1981]. Although they did report stability problems, they attributed them to the integral action of the controller (dynamic instability). However, An and Hollerbach [1987b] later suggested that there were other sources of instability associated with hybrid control that were fundamentally kinematic in nature. Lipkin and Duffy [1986] suggested still other, "kinestatic" arguments against this approach to the hybrid control problem. All of these arguments are presented and discussed in Chapter 2.

A further account of the theory has been given by Craig [1986]. Generalizations using pseudoinverses and projection matrices (which are based on noninvariant methods) are reported by West and Asada [1985]. Nonetheless, to date, this approach has proved generally unsuccessful despite the continued publication of papers on the technique, including Zhang and Paul [1985], Wedel and Saridis [1988], Yabuta, et al. [1988] and Tarn, et al. [1988], to name but a very few.

Salisbury [1980] suggested an architecture not unlike Raibert and Craig [1981], except that the tangible constraints were not accounted for explicitly. Instead, the programmer had to specify a stiffness matrix so as to accommodate contact with a stiff environment. Task axes parallel to the contact surfaces retained their high stiffness for positional accuracy, while those perpendicular to the constraint surface were compliant. Damping control [Whitney, 1977] similiarly uses a damping matrix for this type of accommodation.

Impedance control [Hogan, 1985, and Kazerooni et al., 1986] is presented as a more general formulation. This method views the manipulator as a mass-spring-dashpot system whose values of inertia-stiffness-damping can be arbitrarily specified, usually by modifying various feedback gains. In addition to the obvious problems associated with configuration dependence, as well as potential joint coupling in actuator space, Whitney [1987] showed that such schemes are characterized by sluggish behavior and poor disturbance rejection properties. This is due to the fact that when these gains are varied from a tuned operating point, the closed-loop poles of the system are moved to new locations. Such ad-hoc gain adjustment almost certainly degrades system performance and can even result in dynamic instability.

Because of the strong configuration dependence, it is extremely difficult, if not impossible, to determine what impedance, and therefore what force, the environment will actually see. As a consequence, some very interesting work has been done to formulate the problem in "operational" space [Khatib and Burdick, 1986, and Khatib, 1987], but experimental results thus far have been limited to a single axis. Furthermore, such explicit formulations require a complete redesign of the controller for each new task, since the controller and the task constraints are inextricably intertwined.

Besides the conceptual issues inherent to the hybrid twist and wrench control problem, there are a number of significant hardware issues. For example, voltage-controlled industrial robots have been described as categorically unsuitable for contact tasks, regardless of the feedback control strategy employed [Koren and Ulsoy, 1982]. Other researchers contend that the modern industrial manipulator, itself, is mechanically unsuitable for more advanced control strategies [An et al., 1988], primarily due to the use of highly geared transmissions. An et al. [1988] have proposed the direct-drive robot architecture as a uniquely suitable alternative for such research, despite the significant limitations of these devices.

Clearly, much work remains, but the potential payoff is enormous. Robot force control offers a dramatic increase in the number and types of potentially economic applications of industrial manipulators. As mentioned, force control is well behind machine vision, both in terms of theory as well as industrial implementation. This may be due, at least in part, to the fact that novel vision developments are typically demonstrated in the laboratory, while correspondingly novel approaches to force control are typically simulated on a digital computer.

While simulation is a valuable developmental tool, it can never replace laboratory demonstration. Indeed, as already observed, if the field of robot control is to evolve into a truly scientific/engineering endeavor, rather than an endless pursuit of ever more esoteric mathematics, then research must routinely incorporate experimental verification. The very suggestion of hypothesis verification through simulation alone is simply unacceptable.

1.3 Principal Contributions

While many papers have been presented on the theory of robot force control, there have been almost no experimental results to validate these theories. This research helps to fill that void. Furthermore, these experimental results served not only to validate this cross-coordinated control strategy, but they also facilitated the discovery and solution of some unforseen problems, such as force limit cycles, as well as fostering additional insight into the real issues of robot force control.

Clearly then, what makes this research relatively unique is its experimental basis. Cross-coordinated control was actually tested and verified on an industrial robot, demonstrating typical industrial tasks. The principal contributions of this research can generally be considered to fall into the following four areas:

(1) A geometrically sound, experimentally verified, constraint formulation strategy was described and demonstrated. This formulation is distinct from the servo control problem, due to the choice of control system architecture.

(2) A practical approach for the experimental analysis of the relevant characteristics of the General Electric P60 industrial robot system, as well as the design of suitable servo control compensators, was presented and demonstrated.

(3) A multifunctional, instrumented, anisotropic, mechanically compliant wrench sensor composed of commercially available components was described and utilized for a variety of representative industrial tasks. This sensor facilitated a new level of real-time, end-effector feedback for industrial robots.

(4) A stable, practical control strategy for hybrid twist and wrench control of a voltage-controlled industrial manipulator was demostrated in the laboratory for representative industrial tasks. Significantly, this strategy required only minor hardware modifications to the commercially available robot control system to achieve this capability.

CHAPTER 2
FUNDAMENTAL CONCEPTS

2.1 Introduction and Objective

This chapter introduces the fundamental concepts underlying the hybrid twist and wrench control of a voltage-controlled industrial manipulator. These include screw theory, kinestatics, stability and dc servomotor fundamentals. The treatment is, of necessity, brief, but additional references are suggested for the interested reader.

The chapter begins with an introduction to line geometry and screw theory. It is appropriate to begin here, since those concepts underpin the entire development of this research. An appreciation of the power and elegance of screw theory is, therefore, absolutely essential.

This is followed by the introduction of the concept of kinestatics. This term [Lipkin and Duffy, 1986] is used to refer to the dual relationship that exists between statics and instantaneous kinematics for rigid bodies. This relationship is immediately obvious to those familiar with screw theory. Kinestatic filtering, which refers to the conditioning of feedback signals based solely on our knowledge of the geometry of the model, is also introduced. That concept largely grew out of the work done by Mason [1981]. Since each of the actuators in a typical industrial robot must respond to both the motion and force specifications, kinestatic compatibility is fundamental.

Equally important, though generally ignored in the literature until recently, has been the question of stability for tangibly constrained end-effectors. The stability

problem decomposes into kinematic and dynamic stability issues [An and Hollerbach, 1987a and 1987b] as well as what may be termed "kinestatic" stability issues [Lipkin and Duffy, 1986]. All are discussed.

Also important is an understanding of some of the operating characteristics of dc servomotors, like those used on the GE P60 industrial robot. In recent years, these motors have come to dominate the small to medium size robot actuator market. Conventional control architectures are reviewed [Koren, 1985, and Snyder, 1985].

As previously noted, these topics can only be briefly treated here. The objective is not to make the reader an expert, but rather to ensure that the fundamental concepts, and related issues, are clearly defined, and that additional references are provided for those interested in a more comprehensive treatment.

2.2 Line Geometry and Screw Theory

The theory of screws is classical [Ball, 1900], but its application to robotics can provide new and valuable insights into both the kinematic and static force analysis required. Thus, the instantaneous properties of manipulators are very useful for force and motion control. It should be noted that the joints of industrial manipulators are usually revolute or prismatic. The constraint provided by a revolute joint on adjoining links is determined by the location and direction of the joint axis. However, for a prismatic joint, motion of adjacent links is constrained to be parallel to, or in the direction of, the slider axis. While no unique joint axis exists, it is often convenient to assume that the joint axis lies along the slider. In either case, the kinematic constraint is characterized by a line, thus suggesting the utility of a geometry for manipulator kinematics based on lines, rather than points [Roth, 1984]. Screw theory provides just such a geometry.

Also, an important requirement for many advanced control strategies is the capability for real time velocity computation. While the projective transformations of homogeneous point coordinates gives correct results and continues to be the conventional method for displacement analysis, obtaining closed form solutions for the end-effector has proven difficult. This is due to the necessity to differentiate the projective transformation equations, which only have information about points. Screw coordinates, on the other hand, imply both the position and direction of a line. Thus, they conveniently characterize the velocity of a rigid body, as well as the constraint forces and moments acting on that body [Sugimoto and Matsumoto, 1984].

The chapter begins with an introduction to the mathematical tools necessary to quantitatively describe the spatial motions of a rigid body and the constraint loads imposed by contact with a rigid environment. Although we develop relatively simple screw systems here, their power to provide a simple yet elegant means of understanding the relationship between the geometry of static constraint and instantaneous motion will be obvious. A much more thorough treatment of screw theory may be found in Waldron [1969], Hunt [1978] and Ohwovoriole [1980].

The first step is to define a line segment, which can be thought of as the join of two points. Every line segment lies along an unlimited line, that is, a line that extends to infinity in both directions, as shown in Figure 2.1. The goal is to describe this line segment as a dual-vector, $(\underline{S};\underline{S}_O)$, where $\underline{S}$ and $\underline{S}_O$ are each elements of $\mathbb{R}^3$.

A general line segment may be thought of as having a direction vector, $\underline{S}$, and an origin-dependent vector, $\underline{S}_O$. The direction vector may be determined as the difference of the two point vectors that define the line segment,

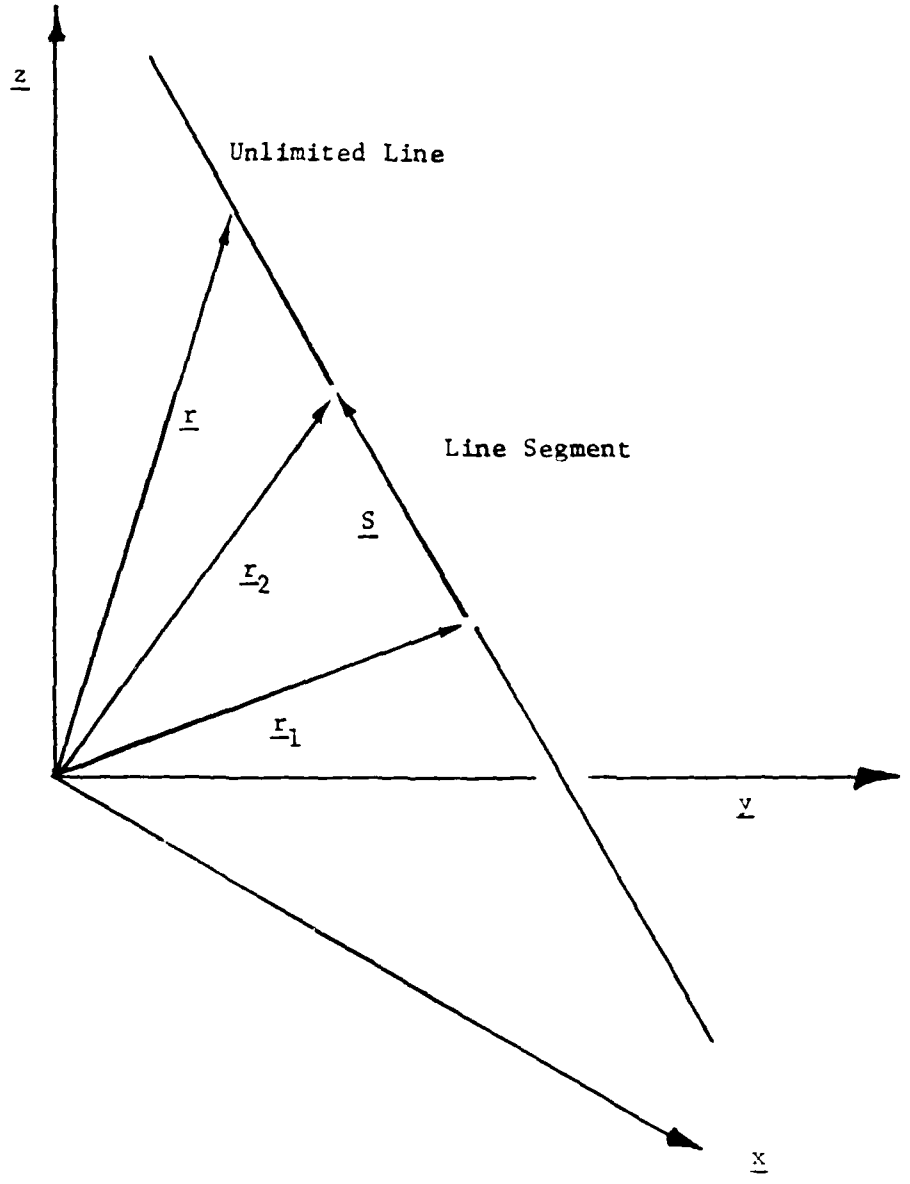


Figure 2.1

$$\underline{S} = \underline{r}_2 - \underline{r}_1 \tag{2.1}$$

where the magnitude of $\underline{S}$ is proportional to the length of the line segment defined by $\underline{r}_1$ and $\underline{r}_2$. Further, the line segment is restricted to lie along the unlimited line. We now define $\underline{r}$ as a vector from the origin to any point on the unlimited line, then

$$(\underline{r} - \underline{r}_1) \times \underline{S} = \underline{0} \tag{2.2}$$

$$\underline{r} \times \underline{S} = \underline{r}_1 \times \underline{S} \tag{2.3}$$

Using Equation 2.1,

$$\underline{r} \times \underline{S} = \underline{r}_1 \times \underline{r}_2 \tag{2.4}$$

Let us now define this origin-dependent vector, $\underline{S}_O$, as

$$\underline{S}_O \equiv \underline{r} \times \underline{S} \tag{2.5}$$

which is the moment of the line about the origin. Thus, the dual-vector $(\underline{S};\underline{S}_O)$ must satisfy the restriction that

$$\underline{S} \cdot \underline{S}_O = 0 \tag{2.6}$$

Given this restriction, this dual-vector has only five independent parameters. Thus it

does not fully define the line segment, since six independent parameters are necessary to define two points.

The sixth parameter necessary to completely specify the line segment is its location on the unlimited line. Klein [1939] observed that if we moved the line segment along the unlimited line, without changing its magnitude or sense, the coordinates, $(\underline{S};\underline{S}_O)$, would remain unchanged. However, if we changed the magnitude or sense (or both), the resulting dual-vector, $(\underline{S}';\underline{S}'_O)$, will differ from $(\underline{S};\underline{S}_O)$ by a scalar multiple proportional to the change in magnitude and sense made to the line segment. Hence this representation is homogeneous, leaving four independent parameters to describe the unlimited line. Two additional parameters, the location and the magnitude and sense are needed to specify a particular line segment along that unlimited line.

The representation of *lines by means of Plücker line coordinates* is convenient. These are expressed as

$$(\underline{S};\underline{S}_O) \cong (L,M,N;P,Q,R) \tag{2.7}$$

where L, M, and N refer to the direction cosines of the line segment and P, Q, and R refer to the moment of the line about the origin. Observe that the quadratric identity [Hunt, 1978] expressed by Equation 2.6 can now be written as

$$LP + MQ + NR = 0 \tag{2.8}$$

and the magnitude of the unit line segment written as

$$\|\underline{S}\| = \sqrt{L^2+M^2+N^2} \tag{2.9}$$

Equation 2.8 is also referred to as the Klein quadric.

There are two special cases which deserve mention. The first is when the line passes through the origin. Here,

$$\underline{r} = \underline{0} \tag{2.10}$$

and hence the coordinates of the line segment are ($\underline{S}$;$\underline{0}$). The second is when the line lies on the plane at infinity. While the correct expression for the first case is fairly obvious, this is perhaps less so for the line at infinity. By convention, we use ($\underline{0}$;$\underline{S}$) to indicate a line at infinity. The "o" subscript is omitted, since there is no origin dependence in this case. The magnitude, for the line segment at infinity, is

$$\|\underline{S}\| = \sqrt{P^2+Q^2+R^2} \tag{2.11}$$

A concise development of the rationale for this convention is presented by Griffis [1988]. Summarizing, a unitized, dual-vector representation of a line segment which satisfies the quadratic identity for all lines has been developed.

$$\underline{S} \cdot \underline{S}_o = 0 \tag{2.12}$$

How simple it would be if these lines formed a vector space. Unfortunately, such is not the case. When one adds two lines together, the result is not, in general, another line [Brand, 1948]. This can be seen by the fact that the result will not, in general, satisfy the quadratic identity. Rather, this general element of $\mathbf{R}^6$ is called a screw, and is denoted by "$\underline{\$}$".

The term screw derives from the kinematic interpretation of this geometric entity. The motion described by a screw consists of a rotation about the screw axis, and a translation parallel to the screw axis, much like the motion of any point in the body of a nut turning on a screw, as shown in Figure 2.2.

Six quantities $(S_1,S_2,S_3;S_4,S_5,S_6)$ can be used to uniquely describe a screw, though only five are independent. Thus there are ∞^5 screws in space. The translation along the screw axis is characterized by a scalar called the pitch, which has the dimensions of length/radians. The pitch of a screw is given by

$$h = \frac{S_1S_4+S_2S_5+S_3S_6}{(S_1^2 + S_2^2 + S_3^2)} \tag{2.13}$$

Observe that if the line coordinatates of the screw axis are given by (L,M,N;P,Q,R), then

$$L = S_1 \tag{2.14}$$

$$M = S_2 \tag{2.15}$$

$$N = S_3 \tag{2.16}$$

$$P = S_4 - hS_1 \tag{2.17}$$

$$Q = S_5 - hS_2 \tag{2.18}$$

$$R = S_6 - hS_3 \tag{2.19}$$

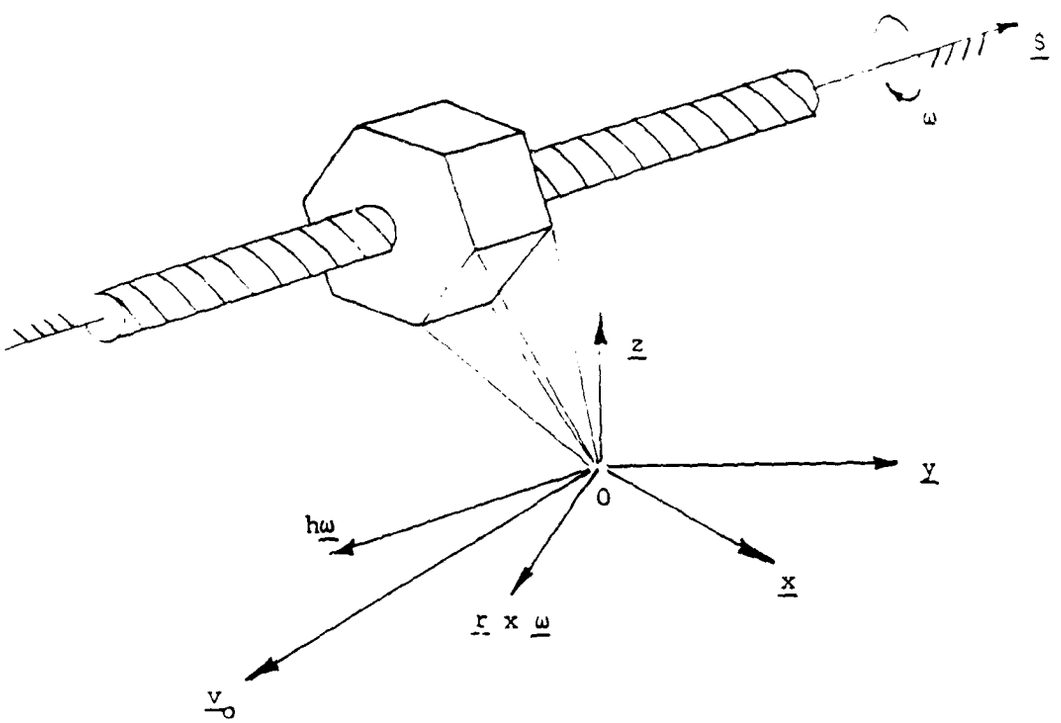


Figure 2.2

An alternative notation for screw coordinates [Hunt, 1978] is to retain the L, M, and N notation, while superscripting the moment terms, P^*, Q^*, and R^*, to differentiate them from line coordinates, P, Q, and R. A table summarizing the screw is presented in Griffis [1988], and is presented here, with permission, for the reader's convenience as Table 2.1.

2.3 Kinestatics

As already stated, the term kinestatics is used to refer to the dual relationship that naturally exists between statics and instantaneous kinematics. In fact, the screw representation emphasizes the underlying dependence of velocities and constraint forces on line geometry. It is well known that the instantaneous velocity of a rigid body may be represented by an angular velocity about a unique line and a translational velocity *parallel to that same line.* Therefore, it is possible to define the instantaneous twist, T, to be a scalar multiple of the unit screw that characterizes the motion. The scalar, ω, called the amplitude of the twist, has the units of radians per second.

$$T = \omega \$ = \omega \begin{bmatrix} L \\ M \\ N \\ P^* \\ Q^* \\ R^* \end{bmatrix} = \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \\ v_{ox} \\ v_{oy} \\ v_{oz} \end{bmatrix} \tag{2.20}$$

When a body's motion is about an instantaneous twist, at that instant it is rotating about that screw axis at ω radians per second and translating along that axis at $h\omega$ units of length per second.

Table 2.1 The Screw Table [Griffis, 1988].

Name	Dual-Vector $\$=(\underline{S};\underline{S}_O)$	Description
Screw	$(\underline{S};\underline{r}\times\underline{S}+h\underline{S})$	$\underline{S}$ is a unit vector of zero dimension. $\underline{S}$ is a direction vector. h is non-zero and finite. $h=\underline{S}\cdot\underline{S}_O$ $\underline{r}$ and h have dimension [L]. $\underline{r}$ is any vector from the origin to the screw's line.
Finitely-Located Line	$(\underline{S};\underline{r}\times\underline{S})$	$\underline{S}$ is a unit vector of zero dimension. $\underline{S}$ is a direction vector. $\underline{S}\cdot\underline{S}_O=0$ $\underline{r}$ has dimension [L]. $\underline{r}$ is any vector from the origin to the line. Special case of screw with $h=0$.
Infinitely-Located Line	$(\underline{0};\underline{S})$	$\underline{S}$ is a unit vector of dimension [L]. $\underline{S}$ is a direction vector. $\underline{S}$ is a free vector. Homogeneous coordinates for a line that lies on the plane at infinity or a screw with infinite pitch.

Instantaneous twists are linear. Further, since they are closed under vector addition and multiplication by a scalar, they form a vector space, albeit of dual-vectors. They may be considered linearly dependent if we can find non-zero scalars, c_1 and c_2 such that

$$c_1\$_1 + c_2\$_2 = \underline{0} \tag{2.21}$$

Thus, it is convenient to use this entity to describe the instantaneous motion of a typical industrial manipulator's end-effector, since it is serially connected to ground by joints which are characteristically rotary or prismatic.

If we specify unit screws to be coincident with these joints, and associate speeds to each, the resulting instantaneous twist at the end-effector can be obtained *via a linear* combination of the joint twists [Lipkin and Duffy, 1982],

$$T = \begin{bmatrix} \underline{S}_1 & \underline{S}_2 & \underline{S}_3 & \underline{S}_4 & \underline{S}_5 & \underline{S}_6 \\ \underline{S}_{O1} & \underline{S}_{O2} & \underline{S}_{O3} & \underline{S}_{O4} & \underline{S}_{O5} & \underline{S}_{O6} \end{bmatrix} \begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \\ \omega_4 \\ \omega_5 \\ \omega_6 \end{bmatrix} \tag{2.22}$$

or

$$T = J\,\underline{\omega} \tag{2.23}$$

where J is a six-by-six matrix called the Jacobian. Observe how conveniently this maps

the time rate of change of the joint positions into the instantaneous twist at the end-effector. The first three rows of the Jacobian are dimensionless, while the last three rows have the dimension of length.

Similarly, it is also well known that a collection of forces and couples on a rigid body can be reduced to a force along a unique line of action and a moment about that line. If we associate a scalar having the units of force, with a unit screw whose axis is that line, the result is called a wrench of intensity f. Thus, one can similarly express the wrench as,

$$W = f\$ = f\begin{bmatrix} L \\ M \\ N \\ P^* \\ Q^* \\ R^* \end{bmatrix} = \begin{bmatrix} f_x \\ f_y \\ f_z \\ m_{ox} \\ m_{oy} \\ m_{oz} \end{bmatrix} \tag{2.24}$$

Given a body in static equilibrium, the sum of the forces and moments about the origin must produce a zero wrench.

Special cases for twists and wrenches are also nicely tabularized by Griffis [1988], and are presented here, with permission and for the reader's convenience, as Table 2.2 and Table 2.3, respectively.

Note that a change in the scalar multiplier, whether the intensity of a wrench, f, or the amplitude of a twist, ω, serves only to change the magnitude of the unit screw. The change in the scalar multiplier leaves the unit screw, with its associated screw axis and its pitch, unchanged. Hence these coordinates, like those of the line presented

Table 2.2 The Twist Table [Griffis, 1988].

Name	Dual-Vector $\omega\underline{\$}=\omega(\underline{S};\underline{S}_O)$	Description
General Twist	$\omega(\underline{S};\underline{r}\times\underline{S}+h\underline{S})=(\underline{\omega};\underline{r}\times\underline{\omega}+h\underline{\omega})$	Scalar multiple of a general screw. ω has dimension [1/T]. Physically represents a body rotating about and translating parallel to a line. The direction vector denotes a relative angular velocity $\underline{\omega}$. The origin-dependent vector denotes the vector sum of a pure translation, $h\underline{\omega}$, and a tangential velocity, $\underline{r}\times\underline{\omega}$.
Rotor or Revolute	$\omega(\underline{S};\underline{r}\times\underline{S})=(\underline{\omega};\underline{r}\times\underline{\omega})$	Scalar multiple of a finitely located line. A twist of zero pitch, a pure rotation. ω has dimension [1/T].
Translation or Prismatic Joint	$\omega(\underline{0};\underline{S})=(\underline{0};\underline{v})$	Scalar multiple of an infinitely located line or a screw with an infinite pitch. A twist that is a pure translation. ω has dimension [1/T]. $\underline{v}$ has dimension [L/T].

Table 2.3 The Wrench Table [Griffis, 1988].

Name	Dual-Vector $f\$=f(\underline{S};\underline{S}_O)$	Description
General Wrench	$f(\underline{S};\underline{r}\times\underline{S}+h\underline{S})=(\underline{f};\underline{r}\times\underline{f}+h\underline{f})$	Scalar *multiple of a general screw.* f has dimension [F]. Combines the effects of a pure force acting along the line $(\underline{S};\underline{r}\times\underline{S})$ and a pure couple acting along the line $(\underline{0};\underline{S})$. The direction vector denotes a force $\underline{f}$. *The origin-dependent vector* denotes the vector sum of the moment of the force, $\underline{r}\times\underline{f}$, and the pure couple , $h\underline{f}$. The line of the wrench does not necessarily intersect the body.
Pure Force	$f(\underline{S};\underline{r}\times\underline{S})=(\underline{f};\underline{r}\times\underline{f})$	Scalar *multiple of a finitely located* line. A wrench of zero pitch, a pure force. f has dimension [F].
Pure Couple	$f(\underline{0};\underline{S})=(\underline{0};\underline{c})$	Scalar multiple of an infinitely *located* line or a screw with an infinite pitch. A wrench that is a pure couple. f has dimension [F]. $\underline{c}$ has dimension [F L].

earlier, are homogeneous. Since screws of finite, non-zero pitch do not satisfy the quadratic identity, there are ∞^5 screws in space. Gibson and Hunt [1988] observed that a screw could be considered to be a point in projective five-dimensional space. This projective five-dimensional space also includes lines, which are special screws.

To be geometrically meaningful, an inner product for screws must be invariant with any translation or rotation of the coordinate system, a change of basis, or a change of scale [Lipkin and Duffy, 1985a and 1986]. Such an inner product is twice Ball's virtual coefficient [Ball, 1900], and is the most fundamental principle of his theory. This inner product for screws is given by the following equivalent expressions:

$$\$_1 \circ \$_2 = \underline{S}_1 \cdot \underline{S}_{O2} + \underline{S}_2 \cdot \underline{S}_{O1} \tag{2.25}$$

$$\$_1 \circ \$_2 = \underline{S}_1 \cdot (\underline{r}_2 \times \underline{S}_2 + h_2\underline{S}_2) + \underline{S}_2 \cdot (\underline{r}_1 \times \underline{S}_1 + h_1\underline{S}_1) \tag{2.26}$$

$$\$_1 \circ \$_2 = -a_{12} \sin\alpha_{12} + (h_1 + h_2) \cos\alpha_{12} \tag{2.27}$$

where a_{12} is the mutually perpendular distance between the screw axes and α_{12} is the angle between them, as shown in Figure 2.3. Alternately, this relationship may be expressed by

$$\$_1 \circ \$_2 = L_1P_2^* + M_1Q_2^* + N_1R_2^* + P_1^*L_2 + Q_1^*M_2 + R_1^*N_2 \tag{2.28}$$

The dimension of this scalar product is length. When $\$_1 \circ \$_2 = 0$, the pair is said to be reciprocal, hence the term "reciprocal product". It is important to note that

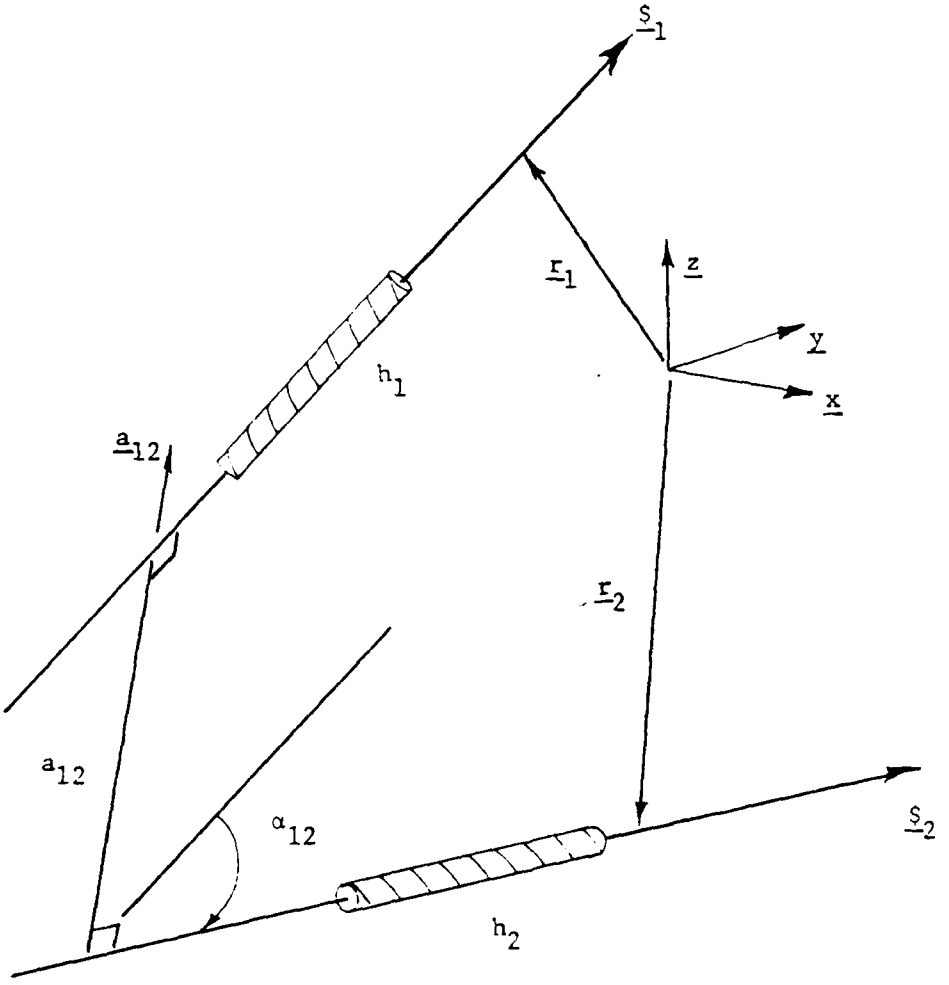

$\underline{s}_1$
$\underline{r}_1$
$\underline{z}$
$\underline{y}$
$\underline{x}$
h_1
$\underline{a}_{12}$
$\underline{r}_2$
a_{12}
α_{12}
$\underline{s}_2$
h_2

Figure 2.3

reciprocal screws will remain reciprocal, even after rotations and translations of the coordinate system, a change of basis, or a change of scale.

Recall that a twist and a wrench can each be represented as a scalar multiple of a unit screw. In fact, the relationship between instantaneous kinematics and statics is well known [Hunt, 1978], and is referred to in this paper as kinestatics [Lipkin and Duffy, 1986].

Consider a wrench, $f\underline{\$}_1$, and a twist, $\omega\underline{\$}_2$. The instantaneous power produced by the wrench acting on the twist is given by the following expression

$$P = f\underline{\$}_1 \circ \omega\underline{\$}_2 = f\omega(\underline{S}_1 \cdot \underline{S}_{O2} + \underline{S}_2 \cdot \underline{S}_{O1}) \tag{2.29}$$

If the unit screws are reciprocal, no power *will be generated, regardless of the amplitude* of the twist or the intensity of the wrench (assuming both are non-zero). By specifying a reciprocal relationship between the instanteous twist and the applied wrench, the vanishing expression for virtual power means that a body in equilibrium will remain in equilibrium. Again, this equilibrium is based solely on geometry. This relationship, rather than any concept of orthogonality, forms the basis of the geometric reasoning behind the constraint formulation strategy of cross-coordinated control, as developed in Chapter 4.

Two interesting concepts with regards to the instantaneous power produced by a wrench acting on a body were studied by Ohwovoriole and Roth [1981]. These other possibilities were termed "repelling" and "contrary" screws. If the virtual power from Equation 2.29 is positive, the screws are said to be repelling, which means that the two bodies in contact tend to move apart. When the virtual power is negative, they are

called contrary, and the bodies tend to move into each other. The so-called jamming condition during assembly will usually occur when screws are contrary. While repelling screws would be permissible under kinestatic constraint, they too are undesirable, as they tend to preclude successful task completion. Only reciprocal screws maintain kinestatic constraints.

It is worth noting that for the case of two planar, mutually perpendicular degrees of freedom, the reciprocal product can reduce to what appears to be orthogonal one,

$$P = f_x \cdot v_x + f_y \cdot v_y \tag{2.30}$$

where f and v refer to forces and translational velocities, respectively, and the subscripts define direction. This explains the success of orthogonal projection in force controlling a single Cartesian axis, while controlling the translational velocity in the direction perpendicular to the commanded force [Stepien, et al., 1987]. However, it may also suggests why similar success has not been reported for controlling additional task axes.

Other examples of wrenches which are reciprocal to twists can be deduced by examining the vanishing of Equation 2.27

- the lines of a force and a rotor intersect, or are parallel.
- the direction vector of a pure couple is perpendicular to the direction vector of the line of a rotor, or
- any pure couple with any pure translation.

Clearly, such wrenches constitute reciprocal constraints.

Kinestatics, then, is the quantification of the permissible twists, called twists of freedom, and the reciprocal constraint wrenches, called wrenches of constraint. By convention [Lipkin and Duffy, 1986], we denote the screw space containing the wrenches of constraint by the matrix A, and the screw space denoting the twists of freedom by a matrix B. The concept of kinestatic filtering has rested on formulations based on these two screw spaces, viz. how can the feedback signals be filtered, based on the kinestatics of the problem, such that the commands sent from the controller are consistent? Two methods are presented in Section 2.4.3. However, to date, these have proven to be limited in application.

A new approach, based on two additional screw spaces, has recently been suggested by Griffis [1988]. This new formulation makes use of reciprocal screw products and knowledge of the physics of the problem to generate two additional screw spaces. These are the twists of non-freedom, space C, which complements the twists of freedom, B, and the impulsive wrenches, D, which form the linear compliment of space A, the wrenches of constraint.

$$\mathbf{R}^6 = \mathrm{A} \oplus \mathrm{D} \tag{2.31}$$

$$\mathbf{R}^6 = \mathrm{B} \oplus \mathrm{C} \tag{2.32}$$

Spaces A and D can thus be used to decompose a sensed wrench, while B and C can be used to decompose a sensed twist. Although promising, this approach has not as yet been implemented in the laboratory.

One reason that the method of Raibert and Craig [1981] has not been successful is that their complementary screw spaces to the twists of freedom, B, and the wrenches of constraint, A, are based on an orthogonal complement, where the inner product is the traditional inner product for vectors. This formulation may work for the two-dimensional, planar problem as already described, but for realistic, industrial tasks, this formulation results in a positive definite expression,

$$\$_1 \cdot \$_2 = S_1 \cdot S_2 + S_{O1} \cdot S_{O2} \tag{2.33}$$

which, unfortunately, is not dimensionally homogeneous. As such, it is inconsistent and dependent on choice of scale.

This formulation is also origin-dependent. Mason specified an appropriate origin as the "center of compliance", a point through which pure forces and pure couples, respectively, produce pure translations and pure rotations, in the same directions. This choice of origin is not based on any physical laws. Furthermore, there is no guarantee that, for a given problem, such an origin will exist at all, or that if it does, it is unique.

Lipkin and Duffy's [1984] complemetary screws spaces were also orthogonal complements, but where the inner product was Ball's reciprocal screw product. Thus, their complementary screw spaces are reciprocal to A and B, respectively. Unfortunately, there is no guarantee that the wrenches of constraint together with their complementary screw space will span $\mathbb{R}^6$. When they don't span $\mathbb{R}^6$, the approach fails. Note that both of these approaches are formulated purely on the geometry, without regard to the physics of the problem.

Therein lies the strength of the new approach by Griffis [1988] in that, the inner product remains the reciprocal screw product, but the formulation ensures that A and D span $\mathbf{R}^6$, as do B and C, respectively. The only disadvantage is that the problem formulation is fairly involved, resulting in a fairly significant computational burden for the general problem.

The essence of the cross-coordinated control method developed here is to circumvent much of this computation by making use of the geometric insight provided by screw theory, together with a physical understanding of the task, in order to model the problem in such a way that filtering is not explicitly required. In this way a strategy is developed using reciprocal screws for the constraint formulation, which effectively decouples the twist and wrench control problems. As long as the commands are reciprocal, *the steady state force and motion commands will never be contradictory*, regardless of which origin, scale, or basis one uses to formulate the problem.

2.4 Stability of Constrained Motion

An essential characteristic of any control system is stability. Under closed loop control, commands will generate responses which, in turn, give rise to new commands. Hence, under these circumstances, it is quite possible for instability to occur.

Historically, most force control strategies have followed the closed loop architecture depicted in Figure 2.4. In general, the robot is commanded along some nominal trajectory, which is then modified by motion updates provided by the force control strategy. This control process has evolved because modern industrial robots are essentially positioning devices.

One cause for instability is that force control can be shown to be very high gain position feedback. This alone, however, may not cause instability were it not for the

GENERALIZED FORCE CONTROL BLOCK DIAGRAM

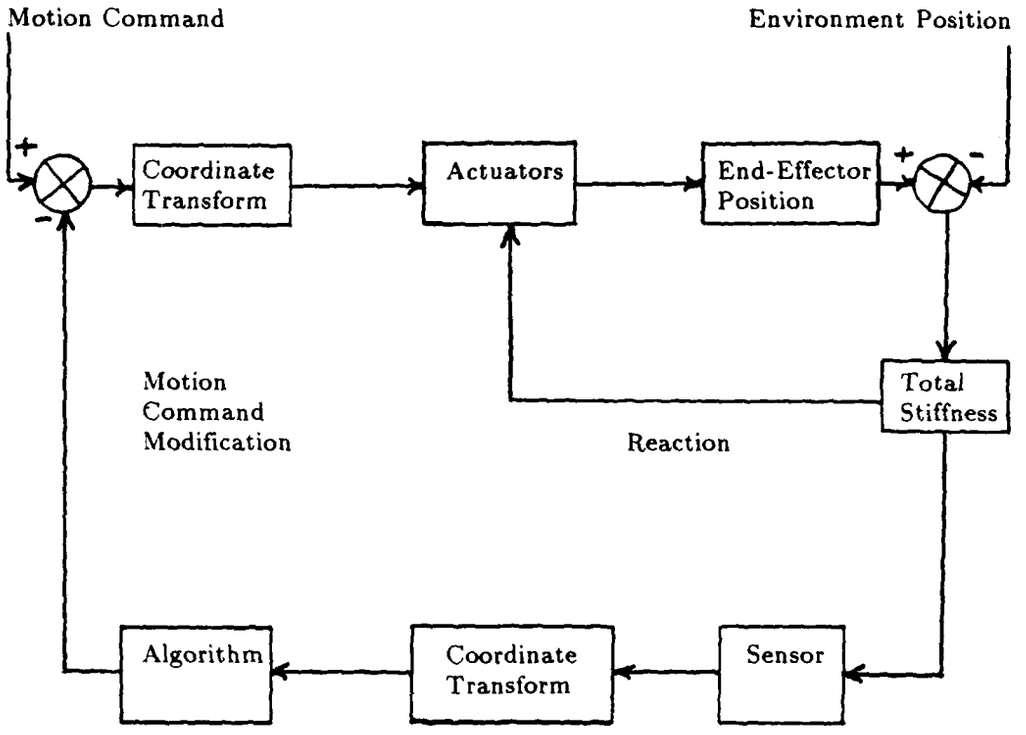


Figure 2.4

additional dynamics introduced by the non-colocation of the sensor and actuator. This type of instabilty is referred to in the literataure as dynamic instability [An and Hollerbach, 1987a].

A second type of instability arises for multi-joint manipulators under certain Cartesian-based control schemes. An and Hollerbach [1987b] refer to this phenomenon as kinematic instability. This type of instability is primarily due to the inverse kinematic transform in the feedback path, but is not limited to points of singularity.

A third type of instability, called kinestatic instability, is due to the incompatibility of force/motion command resolution. The idea was first suggested by Lipkin and Duffy [1985a and 1986]. Essentially, it accounts for unpredictable command signals resulting from strategies that are not invariant with respect to changes of origin, basis or scale.

2.4.1 Dynamic Stability

Robot force controllers tend to go unstable when in contact with a rigid environment [Whitney, 1987] such as one would encounter in an industrial application. This instability is characterized by the end-effector bouncing back and forth uncontrollably against a surface, as reported in experiments by Wedel and Saridis [1988]. As stated earlier, the problem is largely attributable to the high positional gain associated with force feedback.

To illustrate the problem, consider a simple, one degree of freedom model, as shown in Figure 2.5. Ignoring robot dynamics, we model the contact with the environment as a pure stiffness, k_e. This is the effective stiffness due to contact, and includes the effects of stiffnesses due to the robot, the sensor and the workpiece. The sensed force fed back to the force controller is then

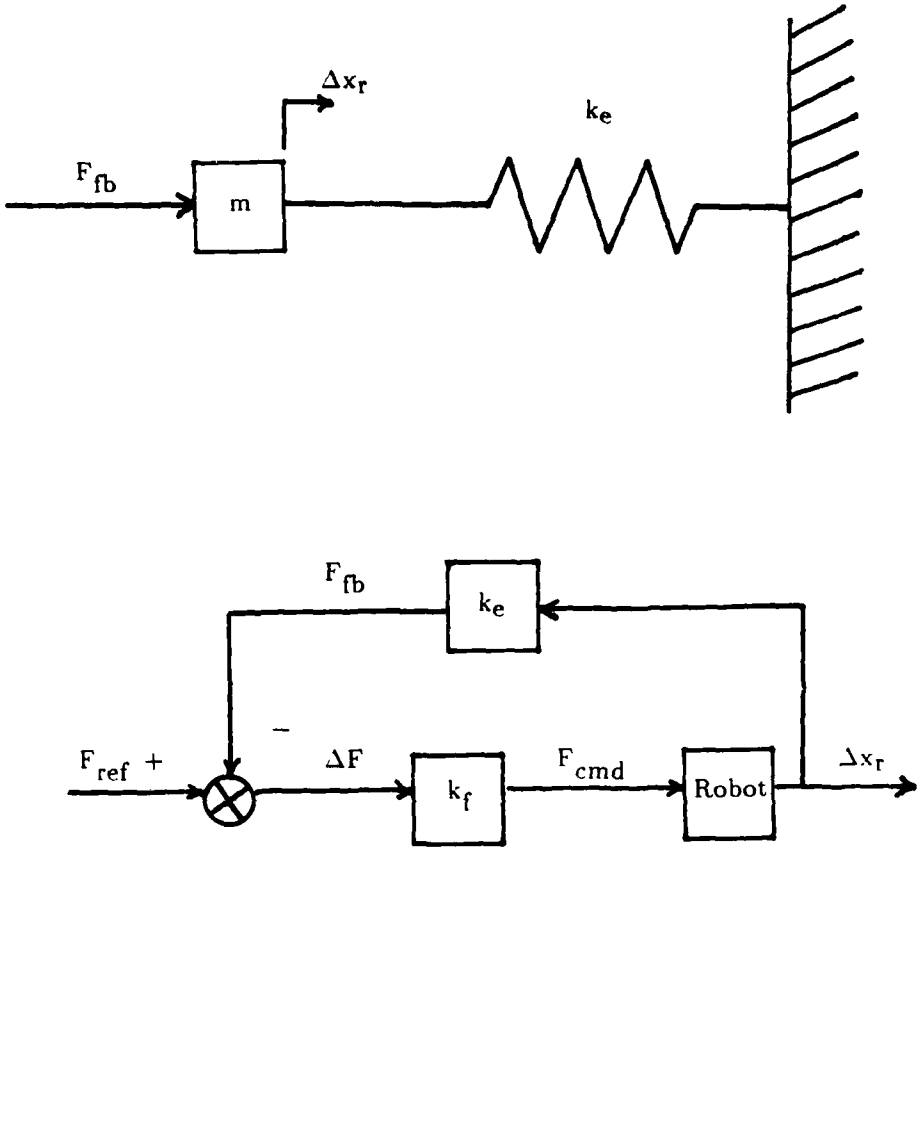


Figure 2.5

$$F_{fb} = k_e \cdot \Delta x_r \tag{2.34}$$

where Δx_r is the change in the position of the mass, m. Assuming a simple, proportional controller with a gain of k_f, the force command signal seen by the actuator, F_{cmd}, is

$$F_{cmd} = k_f (F_{ref} - k_e \cdot \Delta x_r) \tag{2.35}$$

where F_{ref} is the reference value of the force. The pertinant feature here is that total, positional feedback gain is the product of the environmental stiffness, k_e, and the controller gain, k_f.

A more realistic model is shown in Figure 2.6. The open-loop dynamics for such a system are conveniently shown in the Laplace domain by the transfer function

$$\frac{X(s)}{F(s)} = \frac{1}{m_r s^2 + (b_r + b_e)s + k_e} \tag{2.36}$$

where X(s) is the mass displacement, F(s) is the applied force, m_r is the mass of the robot, b_r is a robot damper term to account for viscous damping and b_e the environmental damping term, while k_e accounts for the environmental stiffness.

If we again implement the proportional controller of Equation 2.35, the resulting transfer function is

$$\frac{F_{cmd}(s)}{F_{ref}(s)} = \frac{k_f k_e}{m_r s^2 + (b_r + b_e)s + k_e(1 + k_f)} \tag{2.37}$$

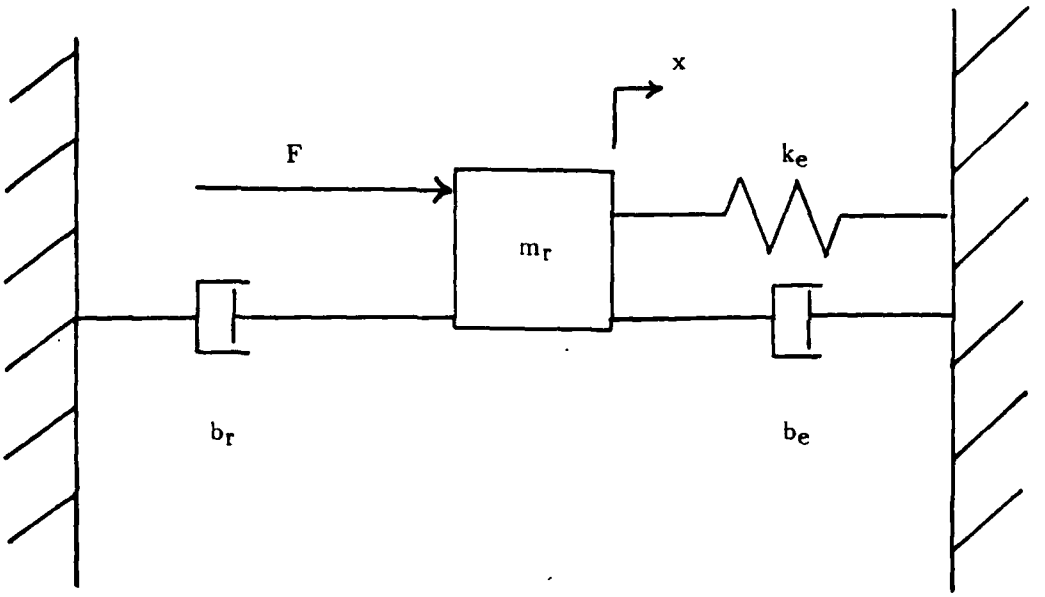


Figure 2.6

It should be noted that the closed-loop gain is proportional to the stiffness of the contact. The characteristic equation is only modified in the stiffness term.

The root locus diagram for this system is shown in Figure 2.7. Increasing the force control gain, k_f, or the stiffness, k_e, will certainly reduce the damping ratio and increase the natural frequency. However, regardless of how much those value are increased, the system will not become unstable. It is well known that is not the case for the actual robot, therefore the second order model is inadequate for the purpose of examining stability [Eppinger and Seering, 1986].

Using two masses for the robot, in order to include both the rigid-body mode and the first vibratory mode of the arm, results in a model as shown in Figure 2.8. This two mass model has two position variables, x_1 and x_2, which correspond to the two mass displacements of the robot. The corresponding open loop transfer functions are

$$\frac{X_1(s)}{F(s)} = \frac{m_2s^2+(b_e+b_2)s+(k_e+k_2)}{d(s^4)} \tag{2.38}$$

$$\frac{X_2(s)}{F(s)} = \frac{b_2s+k_2}{d(s^4)} \tag{2.39}$$

where

$$d(s^4) = [m_1s^2 + (b_1+b_2)s + k_2) \times (m_2s^2 + (b_2+b_e)s + (k_2+k_e)] - (b_2s + k_2)^2 \tag{2.40}$$

The feedback force is still as shown in Equation 2.34, with x_2 replacing x_r, while the feedback law remains as shown in Equation 2.35, with the same substitution.

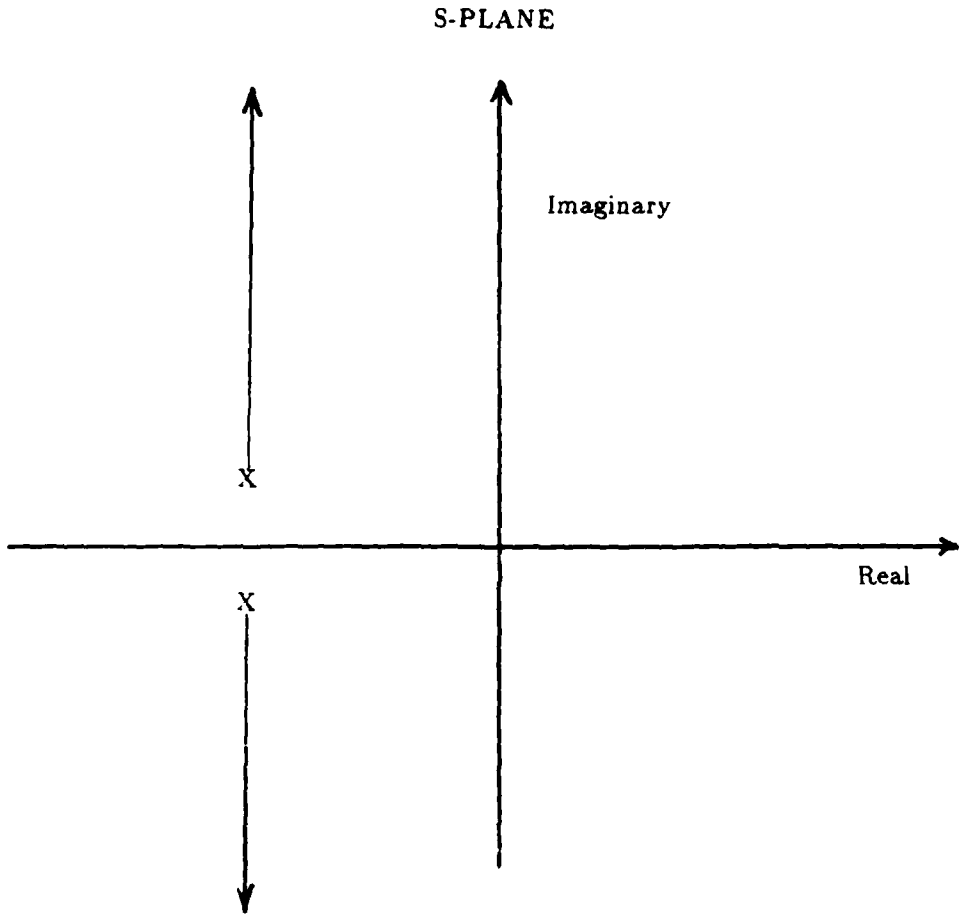


Figure 2.7

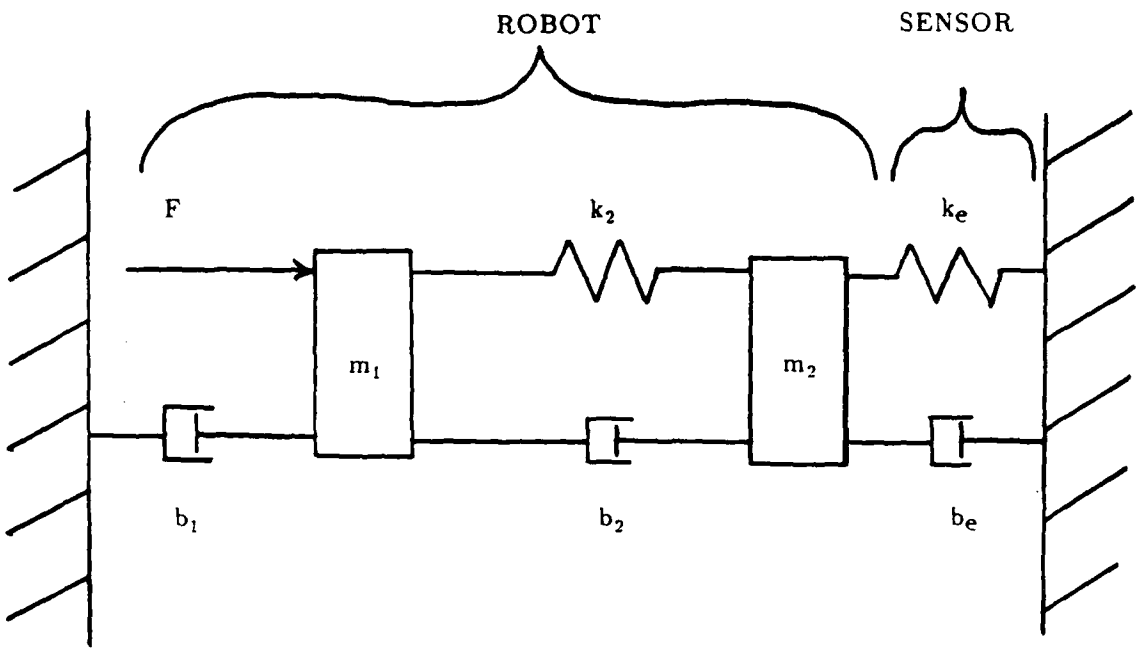


Figure 2.8

The corresponding root locus for this model is shown in Figure 2.9. Observe that the system goes unstable once the system gain exceeds some critical value. This is the behavior observed with actual robots operating under closed loop force control.

Given a system transfer function from Equation 2.37, there are two options to decrease the system gain. The first is to lower the controller gain, k_f. Though this results in sluggish response, with poor disturbance rejection characteristics [Whitney, 1987], this is the approach most often suggested. The alternative is to lower the effective value of k_e. This can be done in a number of ways. Whitney [1987] and Roberts [1984] have both suggested the use of compliant sensors. Problems associated with this approach include bandwidth limitation [Epinger and Seering, 1987] and loss of positional accuracy. The other suggestion is to use compliant coverings [Alberts, 1986], although this approach may not be suitable for all applications. Engineered, mechanically compliant devices, such as the Remote Center Compliance (RCC) device [Whitney and Nevins, 1978], represent another alternative for reducing the value of k_e.

Clearly, the effective loop gain, $k_f k_e$, must be less than the critical gain for closed loop stability. Thus, if the servo is to be tuned to its optimal performance, the question is not whether to introduce compliance, but where? This issue is discussed further in Chapter 5.

The addition of more masses to the model will not necessarily improve the accuracy. As Eppinger and Seering [1987] showed, the addition of mass-spring-dashpot systems to the left of the actuator (see Figure 2.10) or to the right of the sensor, will add poles and zeros to the overall system transfer function in equal numbers, as shown in Figure 2.11. This means that there will be no net effect on the number of asymptotes. The dynamics between the actuator and the sensor, however, add more

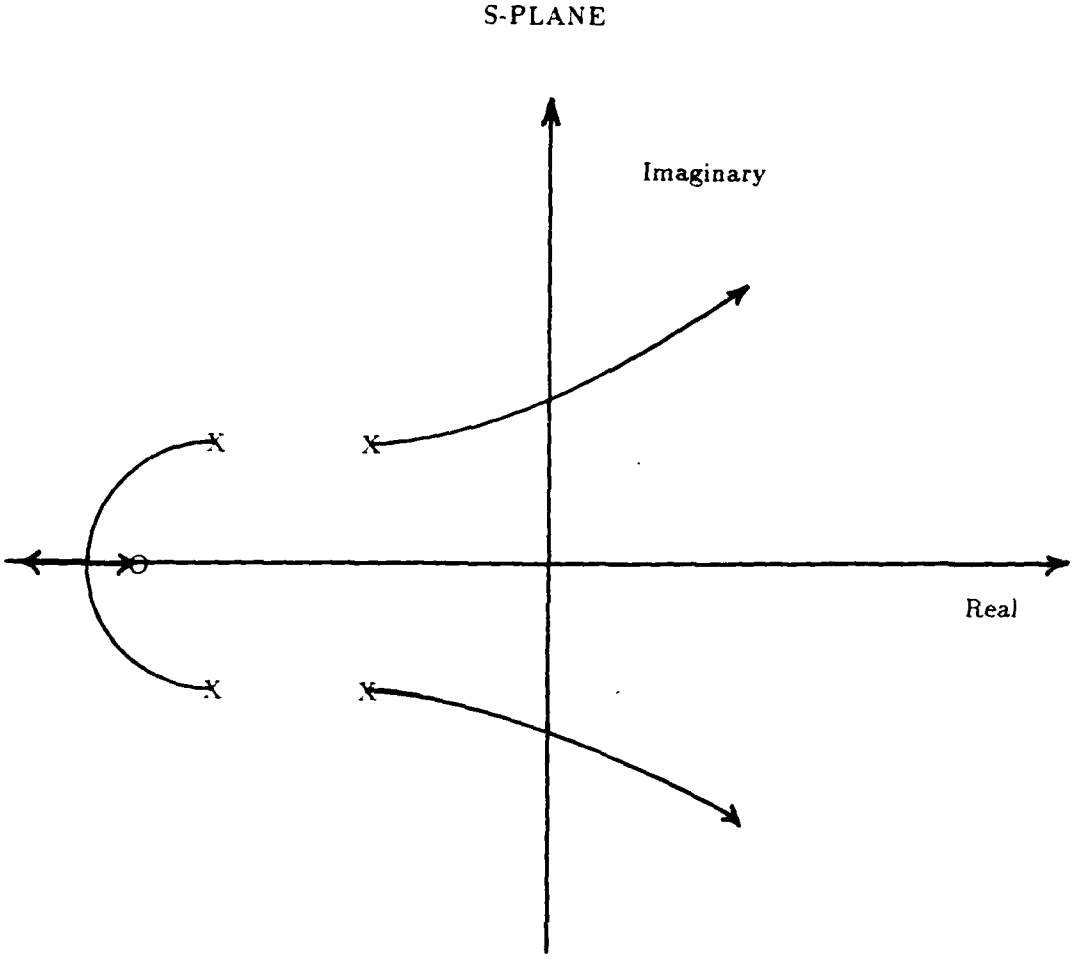


Figure 2.9

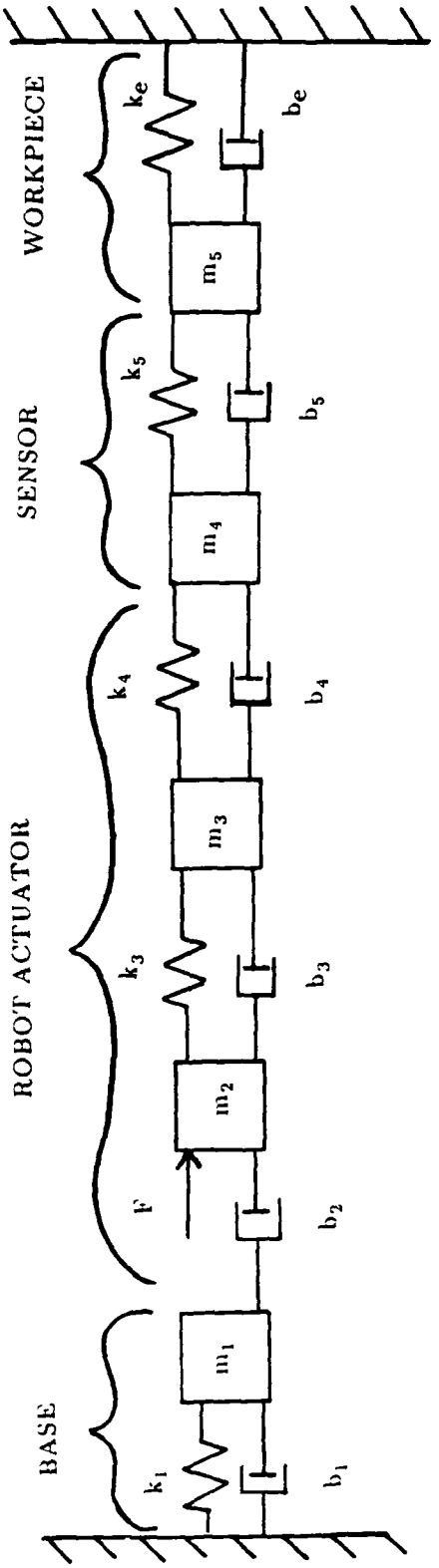

BASE
ROBOT ACTUATOR
SENSOR
WORKPIECE
k_1
b_1
m_1
b_2
F
m_2
k_3
b_3
m_3
k_4
b_4
m_4
k_5
b_5
m_5
k_e
b_e

Figure 2.10

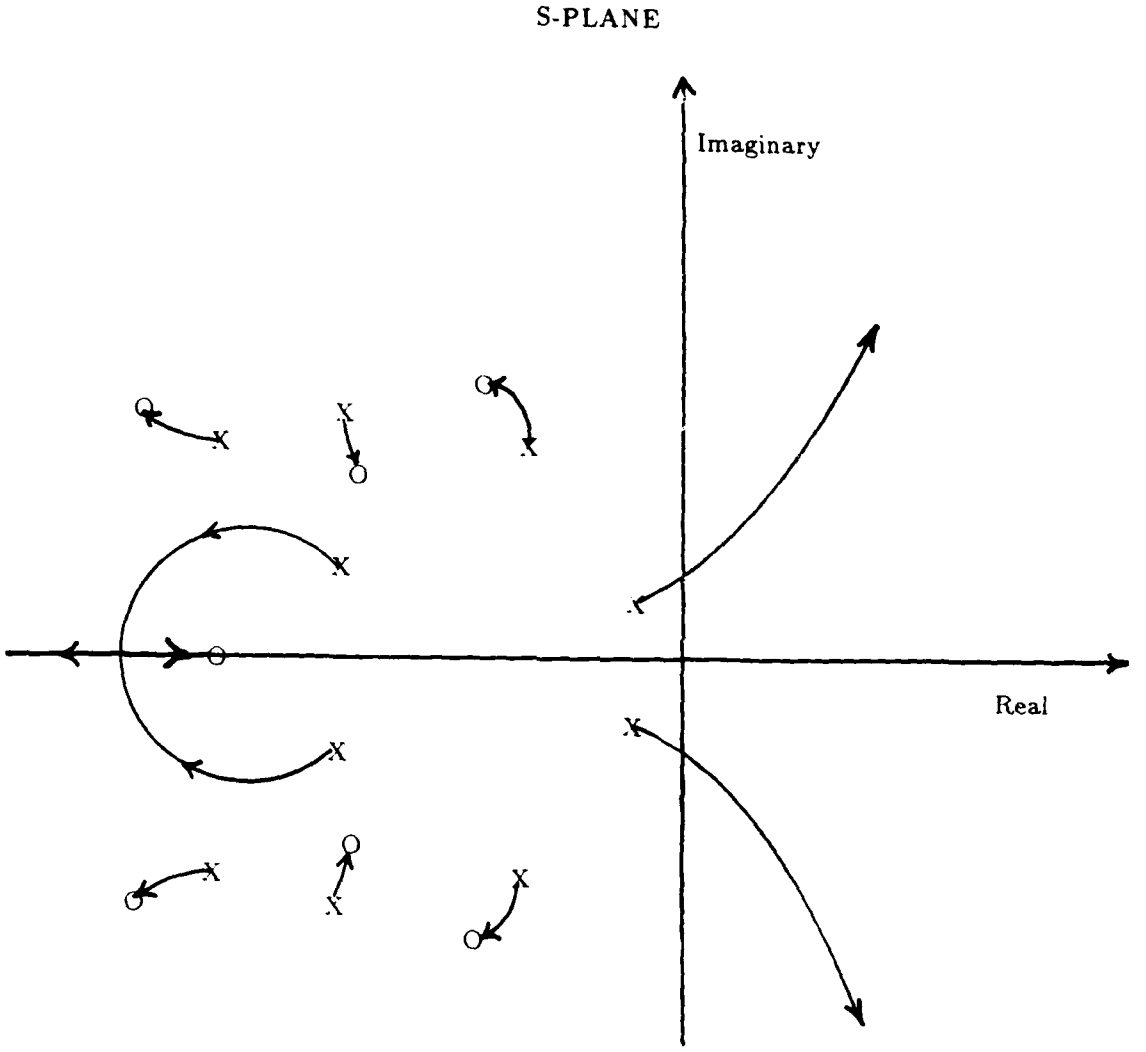


Figure 2.11

poles than zeros. It is precisely these dynamics, due to the non-colocated sensor and actuator, which tend to cause dynamic instability, as the loop is closed by the force control law.

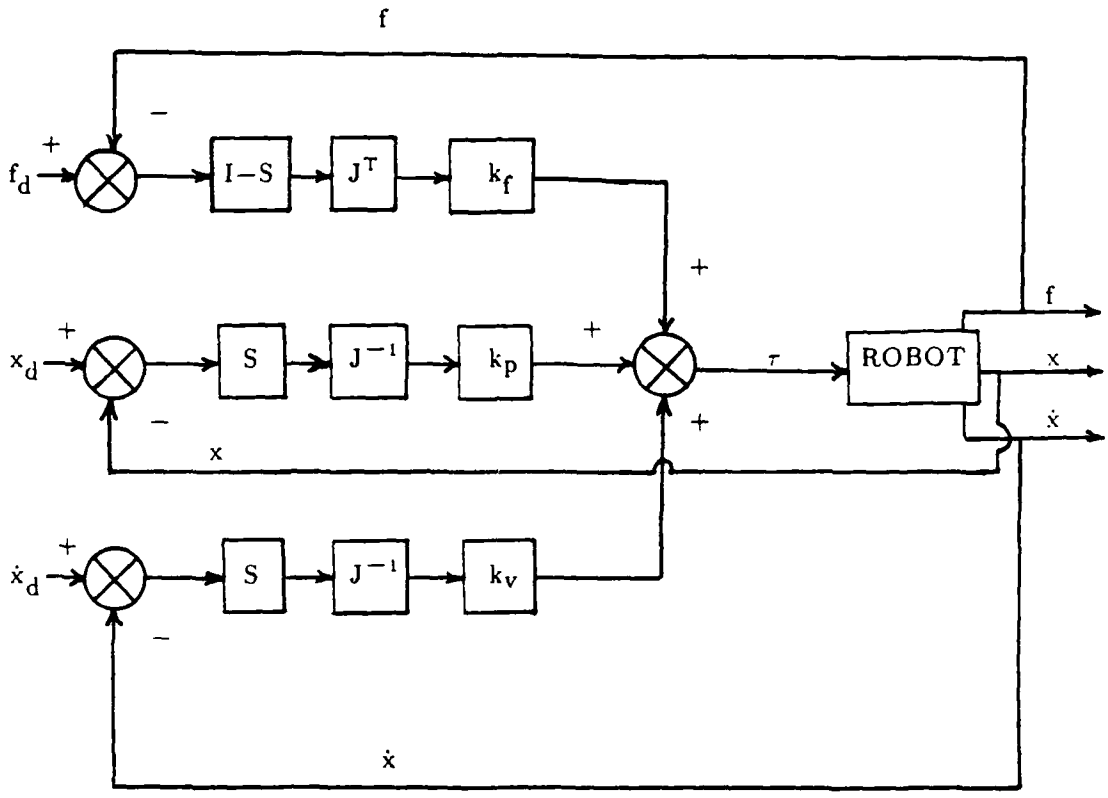
2.4.2 Kinematic Stability

The term "kinematic stability" was introduced by An and Hollerbach [1987b] to refer to instabilities associated with the kinematic structure and configuration of a robot manipulator under certain Cartesian space control strategies, and in particular, the so-called "hybrid control" of Raibert and Craig [1981].

While dynamic instability may occur in robots with either single or multiple joints, kinematic instability is associated with the inverse kinematic transform used in multiple joint robots, using Cartesian based control. These strategies have the advantage that the task geometry can generally be more easily decomposed into motion-controlled variables and force-controlled variables [Mason, 1981]. However, using these variables necessitates a kinematic transform to joint displacements. The major classes of multi-axis, Cartesian based control are summarized below:

- Hybrid Control - This strategy is largely based on the idea of using a selection matrix, S, to decompose the force and position feedback signals, as shown in Figure 2.12. This selection matrix is diagonal, with ones or zeros corresponding to whether the signal component is to be position controlled or force controlled, respectively.
- Stiffness Control - This strategy seeks to control the stiffness in various "directions", so as to achieve the desired contact force or motion as desired. See Figure 2.13.

HYBRID CONTROL

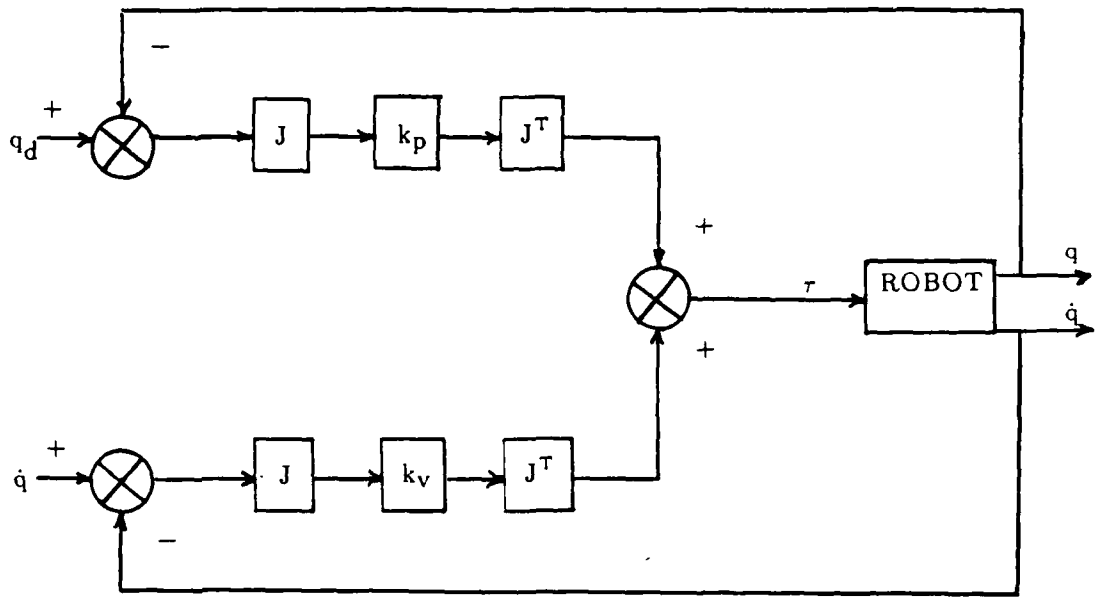


Control Law:

$$\delta\tau = k_p J^{-1} S(x_d - x) + k_v J^{-1} S(\dot{x}_d - \dot{x}) + k_f J^T (I - S)(f_d - f)$$

Figure 2.12

STIFFNESS CONTROL



Control Law:

$$\delta\tau = J^T k_p J(q_d - q) + J^T k_v J(\dot{q}_d - \dot{q})$$

Figure 2.13

- Resolved Acceleration - While originally cast for position control [Luh, Walker and Paul, 1980], it has been reformulated to include force control [Shin and Lee, 1985], as shown in Figure 2.14. This approach is important since it represents a class of methods which includes the operational space method [Khatib, 1983 and 1987], and impedance control [Hogan, 1985 and Kazerooni et al., 1986]. The equivalence of these methods is shown in An et al. [1988] and De Schutter [1986]. These methods are characterized by the explicit inclusion of a model of the inverse dynamics.

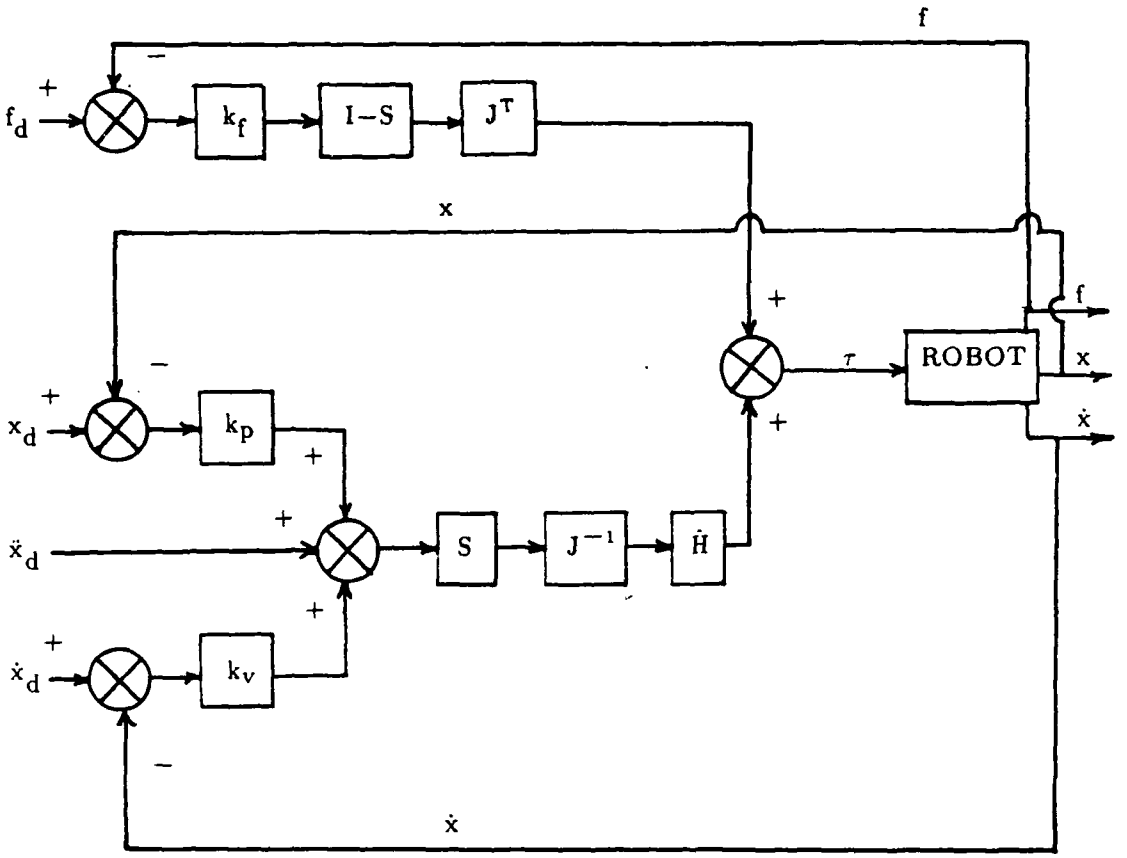
While a Lyapunov method can be applied to analyze global stability [Yabuta et al., 1988], it is sufficient to look at local stability in order to demonstrate the property of kinematic instability. This can be accomplished quite easily by computing the closed-loop eigenvalues of the linealized manipulator system about some operating point, as suggested by An et al. [1988].

Using Newton-Euler, Lagrange, Kane or other approaches, the rigid body dynamics reduces to the following form

$$\tau = H(q)\ddot{q} + C(q,\dot{q}) + G(q) + J(q)^T w \qquad (2.41)$$

In this equation q is a vector of joint displacements, H is the inertia matrix, C is a vector accounting for centrifugal and Coriolis effects, G is a vector which accounts for gravity, and J is the Jacobian matrix relating the vector of joint torques, τ, to the wrench felt at the end effector, w, expressed here in axis coordinates. For contact tasks in which speeds are slow and gravity accounted for elsewhere, Equation 2.41 can be

RESOLVED ACCELERATION CONTROL



Control Law:

$$\delta\tau = \dot{H}J^{-1}S(\ddot{x}_d + k_v(\dot{x}_d - \dot{x}) + k_p(x_d - x)) + J^T(I-S)k_f(f_d - f)$$

Figure 2.14

simplified, leaving

$$\tau = H(q)\ddot{q} + J(q)^T w \tag{2.42}$$

If we consider the most dynamically stable configuration, namely when the end-effector feels zero force, then Equation 2.42 reduces to

$$\tau = H(q)\,\ddot{q} \tag{2.42}$$

The control equation for the hybrid control shown in Figure 2.12 is

$$\delta\tau = k_p J^{-1} S(x_d - x) + k_v J^{-1} S(\dot{x}_d - \dot{x}) + k_f J^T (I - S)(f_d - f) \tag{2.43}$$

where k_p is the positional gain, k_v is the velocity gain, subcript d refers to the desired value while the unsubscripted is the actual value, and S is the orthogonal projection (selection) matrix as described earlier. It is important to note that this equation, as written, is only valid for the two dimensional, planar case. Under this condition, the four elements of the Jacobian matrix all have the units of length. For the general six dimensional problem, the wrench must be modified, as discussed in the next section on kinestatics. This notation, above, is used for simplicity, and to show that kinematic instability can occur, even in the absence of kinestatic instability.

Assuming that $(f_d - f)$ equals zero, one can define the state variable by $\delta y = (\delta q, \delta\dot{q})$, then

$$\delta\dot{y} = \begin{bmatrix} 0 & I \\ 0 & 0 \end{bmatrix} \delta y + \begin{bmatrix} 0 \\ H^{-1} \end{bmatrix} \delta\tau \tag{2.44}$$

If one again neglects any constraint wrench, the control equation becomes

$$\delta\tau = [\ -k_p\ J^{-1}\ S\ J \quad -k_v\ J^{-1}\ S\ J\ ] \begin{bmatrix} \delta q \\ \delta\dot{q} \end{bmatrix} \tag{2.45}$$

The closed-loop system can thus be characterized by

$$\delta\dot{y} = \begin{bmatrix} 0 & I \\ -H^{-1}k_pJ^{-1}SJ & -H^{-1}k_vJ^{-1}SJ \end{bmatrix} \delta y \tag{2.46}$$

or,

$$\delta\dot{y} = A\ \delta y \tag{2.47}$$

where A is called the characteristic matrix of the system. In order to guarantee local stability at some operating point, the eigenvalues of A, which are the closed loop poles of the system, must have negative real parts.

The inertia matrix of a two-link manipulator as shown in Figure 2.15, is given in Brady et al. [1982] as

TWO LINK ROBOT

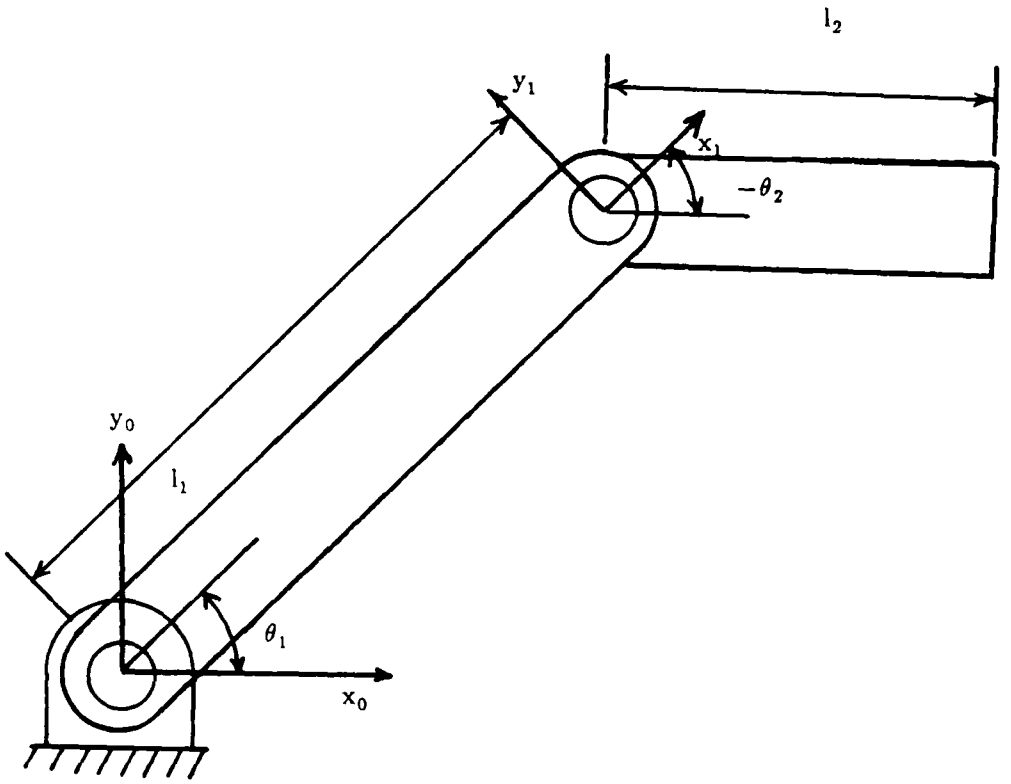


Figure 2.15

$$H(q) = \begin{bmatrix} H_{11} H_{12} \\ H_{21} H_{22} \end{bmatrix} \tag{2.48}$$

where

$$H_{11} = h_1 + h_2 + m_2 l_1 l_2 \cos(q_2) + \tfrac{1}{4}(m_1 l_1^2 + m_2 l_2^2) + m_2 l_1^2$$

$$H_{12} = H_{21} = h_2 + \tfrac{1}{4} m_2 l_2^2 + \tfrac{1}{2} m_2 l_1 l_2 \cos(q_2)$$

$$H_{22} = h_2 + \tfrac{1}{4} m_2 l_2^2$$

and where m_i and h_i are the respective mass and inertia terms about the center of gravity for these two links and l_i the corresponding lengths.

An et al. [1988] examined two cases of the application of hybrid control to this problem. For the first case S=(1,1) diagonal. Not surprisingly, this case is stable, since it is essentially proportional plus derivative (PD) position control. The second case was for S=(0,1) diagonal. Here, the interaction of the inertia matrix, H, with the J^{-1} matrix caused the eigenvalues of A to become positive. This was demonstrated by them both in simulation and experimentally on their direct-drive, experimental robot. Further, when the end-effector was brought into contact with a stiff wall, the robot remained unstable. This meant that the addition of the force feedback term did nothing to stabilize the kinematic instability.

Their experiments showed that smaller feedback gains could reduce the region of instability, but that the inherent instability could not be eliminated. Their work showed that there will always be regions of instability encompassing the points of singularity, with the extent of the regions depending on the magnitude of feedback gains.

Interestingly enough, a two-joint polar manipulator does not experience this problem. The two-link polar manipular was investigated since it characterises the

kinematics of the Stanford Arm, as used by Raibert and Craig [1981]. Simulations by An et al., [1988] showed that a two-joint polar manipulator does not, in fact, display kinematic instability. This result tends to support the contention of Raibert and Craig [1981], namely that the instabilities they observed in the laboratory were in fact due to the integral action of their controller (dynamic instabilities).

Less surprising is that the stiffness control and resolved acceleration control are also apparently immune to this problem. Looking first at resolved acceleration control, the linearized controller can be written as

$$\delta\tau = \begin{bmatrix} -\hat{H}\, J^{-1}\, k_p\, S\, J & -\hat{H}\, J^{-1}\, k_v\, S\, J \end{bmatrix} \begin{bmatrix} \delta q \\ \delta \dot{q} \end{bmatrix} \tag{2.49}$$

where $\hat{H}$ models the dynamics. Comparing this equation with Equation 2.45, we see that the closed loop system can be written in the form

$$\delta\dot{y} = \begin{bmatrix} 0 & I \\ -J^{-1}\, S\, k_p\, J & -J^{-1}\, S\, k_v\, J \end{bmatrix} \delta y \tag{2.50}$$

or,

$$\delta\dot{y} = A\, \delta y \tag{2.51}$$

provided as $\hat{H}$ adequately models the dynamics. The resulting expression can then be written as a similarity transform where

$$A = \begin{bmatrix} J^{-1} & 0 \\ 0 & J^{-1} \end{bmatrix} \begin{bmatrix} 0 & I \\ -S\,k_p & -S\,k_v \end{bmatrix} \begin{bmatrix} J & 0 \\ 0 & J \end{bmatrix} \tag{2.52}$$

$$A = B^{-1}\,D\,B \tag{2.53}$$

Since the eigenvalues of D are preserved under any similarity transform, and since D consists only of S, k_p, k_v, and I, the system is stable based on choices of feedback gains k_p and k_v. No adverse interaction between the inertia matrix and the inverse Jacobian matrix is possible, since the inertia matrix has been eliminated. An et al. [1988] have shown that even with a 50% error in inertia parameters, the system remained kinematically stable.

The analysis is not as straight forward for stiffness control.

$$\delta\tau = [\, -J^T\,k_p\,J, \quad -J^T\,k_v\,J\,] \begin{bmatrix} \delta q \\ \delta\dot{q} \end{bmatrix} \tag{2.54}$$

The corresponding closed-loop system can then be described by

$$\delta\dot{y} = \begin{bmatrix} 0 & I \\ -H^{-1}J^T\,k_p\,J & -H^{-1}J^T\,k_v\,J \end{bmatrix} \delta y \tag{2.55}$$

or,

$$\delta\dot{y} = A\,\delta y \tag{2.56}$$

Since the inertia matrix is not eliminated, it is not immediately clear whether this approach is stable. However, repeated simulation and experimentation by An et al. [1988] did indicate stability. Apparently, employing J^T instead of J^{-1} precludes kinematic instability.

2.4.3 Kinestatic Instability

Kinestatic instability was first suggested by Lipkin and Duffy [1985a]. This instability is a phenomenon associated with the problem of filtering spatial force and motion feedback signals in such a way as to ensure compatability. Lipkin and Duffy [1988] determined that among the necessary criteria to prevent kinestatic instability was the requirement that the decomposition process be invariant with respect to a change of origin, basis, or scale. That is, a change in the parameterization of the problem must not change the physical nature of the problem.

The formulation of kinestatic filtering is based on the unspecified matrices $\tilde{\Psi}$ and $\tilde{\psi}$ [Lipkin and Duffy, 1988]. They suggested that the filtered twist and wrench be decomposed

$$T_B = P_B\,T \tag{2.57}$$

and

$$w_a = p_a\,w \tag{2.58}$$

where

$$P_B = B(B^T\tilde{\Psi}B)^{-1}B^T\tilde{\Psi} \tag{2.59}$$

and

$$p_a = a(a^T \tilde{\psi}\, a)^{-1}\, a^T \tilde{\psi} \tag{2.60}$$

where capital letters imply the use of ray coordinates and lower case letters imply the use of axis coordinates.

Consider, by way of example, the peg in the hole problem as depicted in Figure 2.16. Using $\tilde{\Psi} = \tilde{\psi} = \tilde{\Delta}$, where

$$\tilde{\Delta} = \begin{bmatrix} \bullet & I_3 \\ I_3 & \bullet \end{bmatrix} \tag{2.61}$$

the model of the environment is

$$B = \begin{bmatrix} k & \bullet \\ \bullet & k \end{bmatrix} \tag{2.62}$$

$$a = \begin{bmatrix} i & j & \bullet & \bullet \\ \bullet & \bullet & i & j \end{bmatrix} \tag{2.63}$$

where B are the twists of freedom and a are the wrenches of constraints The symbols i, j, and k represent unit vectors in the coordinate directions and "•" is the zero vector. The corresponding filter matrices are

$$P_B = \begin{bmatrix} \bullet & \bullet & k & \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet & \bullet & \bullet & k \end{bmatrix} \tag{2.64}$$

PEG-IN-THE-HOLE

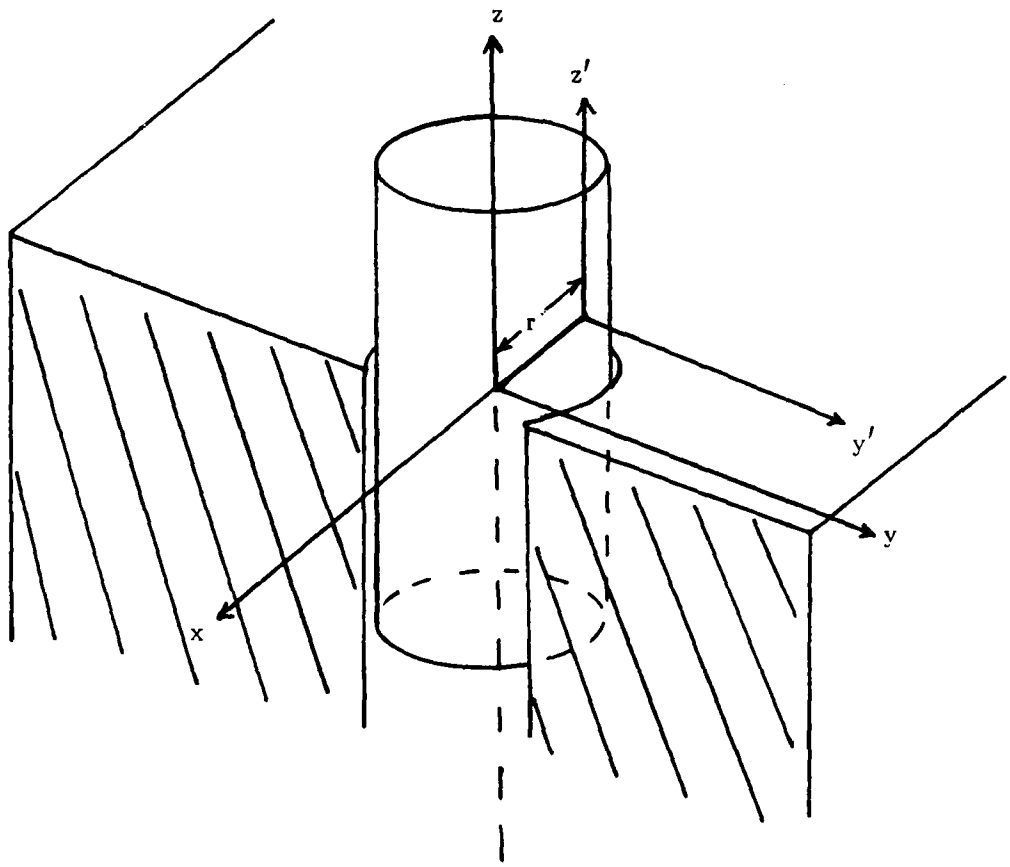


Figure 2.16

$$p_a = \begin{bmatrix} i & j & \bullet & \bullet & \bullet & \bullet \\ \bullet & \bullet & \bullet & i & j & \bullet \end{bmatrix} \tag{2.65}$$

P_B filters the twists except for translations and rotations about the Z axis. The matrix p_a filters wrenches to preclude forces and moments along and about the Z axis, respectively. These are the desired results.

If we consider a unit translation along the Z axis (consistent twist) and a translation along the y axis (inconsistent twist), the results are, respectively,

$$T_1 = \begin{bmatrix} \bullet \\ k \end{bmatrix}, \quad T_{B1} = \begin{bmatrix} \bullet \\ k \end{bmatrix} \tag{2.65}$$

$$T_2 = \begin{bmatrix} \bullet \\ j \end{bmatrix}, \quad T_{B2} = \begin{bmatrix} \bullet \\ \bullet \end{bmatrix} \tag{2.66}$$

The results are correct. Interestingly enough, because of the symmetry, choosing $\tilde{\Psi} = \tilde{\iota} = I_6$, which corresponds to using an orthogonal product as the inner product, also gives correct results.

However, consider what happens when we translate the origin a distance r along the negative X axis, without altering the physical problem in any way. Using $\tilde{\Psi} = \tilde{\iota} = \tilde{\Delta}$, corresponding to the reciprocal product, gives

$$B' = \begin{bmatrix} k & -rj \\ \bullet & k \end{bmatrix} \tag{2.67}$$

$$a' = \begin{bmatrix} i & j & \bullet & \bullet \\ \bullet & rk & i & j \end{bmatrix} \tag{2.68}$$

$$P'_B = \begin{bmatrix} \bullet & \bullet & k & \bullet & -rk & -rj \\ \bullet & \bullet & \bullet & \bullet & \bullet & k \end{bmatrix} \tag{2.69}$$

$$p'_a = \begin{bmatrix} i & j & \bullet & \bullet & \bullet & \bullet \\ \bullet & rk & rj & i & j & \bullet \end{bmatrix} \tag{2.70}$$

$$T'_1 = \begin{bmatrix} -rj \\ k \end{bmatrix}, \quad T'_{B1} = \begin{bmatrix} -rj \\ k \end{bmatrix} \tag{2.71}$$

$$T'_2 = \begin{bmatrix} rk \\ j \end{bmatrix}, \quad T'_{B2} = \begin{bmatrix} \bullet \\ \bullet \end{bmatrix} \tag{2.72}$$

which is correct. Using $\tilde{\Psi} = \tilde{v} = I_6$, which corresponds to the orthogonal product, gives

$$P'_B = \begin{bmatrix} \bullet & r^2dj & k & \bullet & \bullet & -rdj \\ \bullet & -rdk & \bullet & \bullet & \bullet & dk \end{bmatrix} \tag{2.73}$$

$$p'_a = \begin{bmatrix} i & dj & \bullet & \bullet & \bullet & rdj \\ \bullet & rdk & \bullet & i & j & r^2dk \end{bmatrix} \tag{2.74}$$

where

$$d = (1 + r_2)^{-1}$$

$$T'_1 = \begin{bmatrix} -rj \\ k \end{bmatrix}, \quad T'_{B1} = \begin{bmatrix} -rj \\ k \end{bmatrix} \tag{2.75}$$

$$T'_2 = \begin{bmatrix} rk \\ j \end{bmatrix}, \quad T'_{B2} = \begin{bmatrix} rk \\ \bullet \end{bmatrix} \tag{2.76}$$

The first twist, a translation in the direction, is correctly left unchanged. The second twist, a rotation about the y axis which is inconsistent, results in a translation into or out of the hole. Observe that the magnitude and sense of this translation is a function of where we choose the origin.

This, of course, typifies the problem of the non-invariant method, namely that it is possible to get different results by merely changing some way of describing the problem without changing the physical problem. Clearly, this is unacceptable.

The problem is in the use of I_6 to construct the filter. Lipkin and Duffy [1985a] showed that this was actually the metric for elliptical space, that is, that the geometric

interpretation of orthgonality is equivalent to two screws being elliptical conjugates. While orthogonality is invariant with respect to rotations and reflections of a coordinate system, it is not invariant with respect to translations of the origin, as has been demonstrated. Nor is it invariant with respect to a change of scale.

As previously noted, this metric appears to have been mistakenly introduced because of the vanishing of instantaneous power for the two dimensional, planar case, which reduces to what looks like an orthogonal product as shown in Equation 2.30. However, such products as

$$T^T T = \Omega^T \Omega + V^T V \tag{2.77}$$

or

$$w^T w = f^T f + m^T m \tag{2.78}$$

and their "norms" are without Euclidean meaning, and thus should not be used.

2.5 DC Servomotors

In order to better understand how this control strategy was implemented, it is important to understand something about the actuators that were used. Generally, robots have used either pneumatic, hydraulic or electric actuators. Pneumatic and hydraulic actuators are both powered by moving fluids, with pneumatic systems using compressed air and hydraulic systems using pressurized oil. Electric actuators consist primarily of dc and ac servomotors and stepper motors.

Due to recent advances, most small and medium robots, such as the GE P60, use electric motors, and especially dc servomotors. These motors provide excellent controllability with a minimum of maintenance required. An excellent reference on the

theory and practice of dc servomotors can be found in an engineering handbook produced by the Electrocraft Corporation [Rosenblum, 1975]. An interesting aspect of the torque-speed curves will be elaborated upon here, since it is important to the development of this control strategy.

The torque generated by a dc motor can be approximated by

$$T_m = K_m \cdot I_a \tag{2.79}$$

where T_m is the motor torque, I_a is the current flowing through the armature and K_m is the motor torque constant. An important effect associated with the dc motor is the back-emf, or electro-motive force. This is because a dc motor is similar to a dc generator. Thus,

$$e_b = K_b \cdot \omega \tag{2.80}$$

where e_b is the back-emf, K_b is the voltage constant of the motor, and ω is the angular velocity. The effect of this back-emf is to act as a viscous damper.

Supplying a voltage across the motor terminals, E, with a known armature resistance, R_a, the armature curent can then be determined from

$$I_a = \frac{E - K_b \omega}{R_a} \tag{2.81}$$

As the speed of the motor increases, so too does the back-emf. This effect continues until a steady state operating speed is achieved. A block diagram depicting this effect is shown in Figure 2.17, where J is the rotor inertia.

DC MOTOR BLOCK DIAGRAM

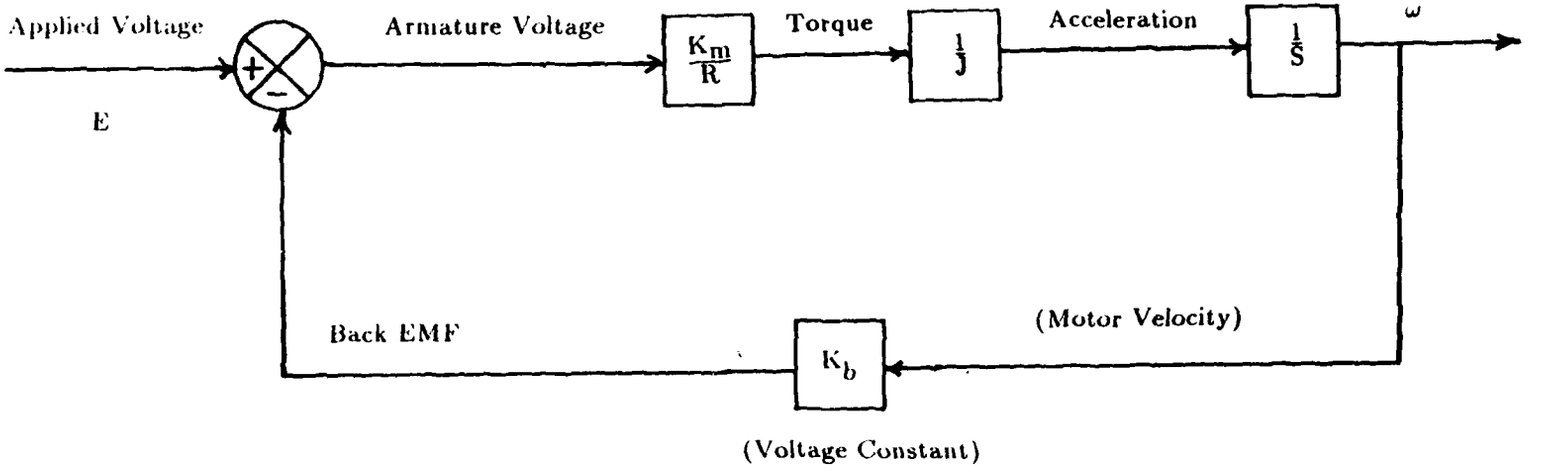


Figure 2.17

For permanent magnet (PM) motors, such as those found on the GE P60, no power is needed for the field structure. Hence, the static flux remains essentially constant, resulting in the familiar torque-speed curves as shown in Figure 2.18.

Each of the straight lines in this figure can be described by the equation

$$\tau = k_1 \cdot E + k_2 \cdot \omega \tag{2.82}$$

where k_1 and k_2 are constants. Note that this equation assumes negligible armature inductance. The constant k_1 relates the applied voltage to torque under stall conditions (ω=0):

$$k_1 = \frac{\text{stall torque at rated voltage}}{\text{rated voltage}} \tag{2.83}$$

The constant k_2 is the slope of the linearized torque-speed curves

$$k_2 = \frac{\Delta\tau}{\Delta\omega} \tag{2.84}$$

This value is often quoted in the manufacturers literature by

$$k_2 = \frac{\text{stall torque at rated voltage}}{\text{no-load speed at rated voltage}} \tag{2.85}$$

Of course, the actual torque-speed curves are not precisely linear, nor are they precisely equidistant. However, over a limited range, the assumption of linearity

REPRESENTATIVE TORQUE/SPEED CURVES

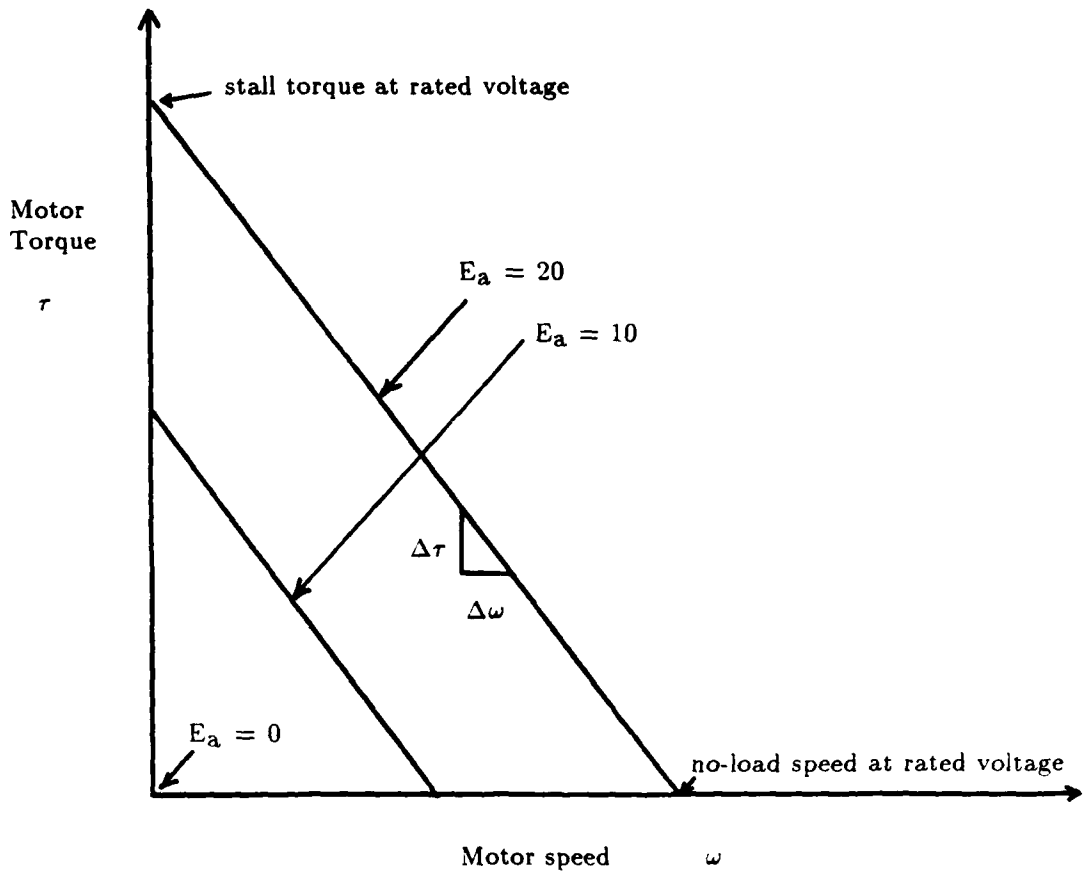


Figure 2.18

facilitates the analysis by more clearly revealing certain qualitative relationships, and is adequate for these purposes.

The torque produced by the motor must drive both the static and dynamic loads associated with the robot arm. Constraint loads applied at the end effector, such as those discussed in this research, fall into the category of static loads.

Conventional wisdom has suggested that there are two alternatives to control these dc motors. These alternatives are to control the torque by use of a current amplifier, or to control the speed by use of a voltage amplifier [Koren, 1985].

Current-controlled motors are said to be able to directly control the motor torque. As such, it is suggested that they are well suited to assembly tasks. Unimates's Programmable Universal Machine for Assembly, or PUMA robot, is such a device. However, if the load torques are not correctly estimated, these devices will display eratic speed performance. When one considers the difficulty in accurately measuring frictional and inertial loads, off-line programming of these devices becomes difficult at best.

Voltage-controlled robots, on the other hand, use a voltage amplifier [Koren, 1985], or pulse-width modulation (PWM) [Snyder, 1985], in the drive unit. PWM, which is used on the GE P60, works by using a rapidly changing control signal. This rate is so fast, as compared to the response speed of the servomechanism, that the net effect, over the operating range, will be a response to only the dc component of the drive signal. The principal advantage of using PWM rather than the equivalent, linear circuit is simplification of the drive system electronics, which greatly simplifies computer interfacing. Unlike current-controlled drive units, variations in the load affect only the time constant of the response, but not the time required to reach the target position. Voltage-controlled servomotors will maintain constant speed, drawing whatever current

is required. As such, without a current limiter in the circuit, contact with a rigid environment could prove disasterous. These robots, such as the GE P60, are normally used in applications where speed control is important, such as spray-painting, welding, or sealant application.

Some have gone so far as to say, "It is clear that we cannot simultaneously control speed and torque, regardless of the types of feedback added to the system" [Koren and Vlsoy, 1982, page 228]. Fortunately, this is incorrect. The control strategy used in this research was implemented on a voltage-controlled, industrial robot, using a feedback strategy that does simultaneously control both the speed of, and the torque produced by, a dc servomotor. This was accomplished by careful formulation of the constraints, such that the net power produced by the reciprocal product of the constraint wrench and the commanded twist was zero. The constraint formulation procedure, which results in the generation of these kinestatically consistent drive signals, is discussed in detail in Chapter 4.

2.6 Results and Conclusions

As stated at the outset, the purpose of this chapter was to introduce certain essential concepts which were fundamental to the development and implementation of this hybrid twist and wrench control strategy, referred to as cross-coordinated control. Additional references were also cited.

Beginning with an introduction to line geometry and screw theory, the reader was motivated to appreciate the elegance of this representation due to the natural characterization of modern robot joints as lines. After presenting the notation, the concept of kinestatics, or the dual relationship between statics and instantaneous kinematics, was introduced. So too was the reciprocal product of screws, and its physical meaning with

respect to instantaneous power. Besides the concept of kinestatic filtering, a new approach to wrench and twist decomposition was reported.

The fundamental prerequisite of all closed-loop control systems, stability, was discussed at length. Dynamic stability, which involves issues relating to the stiffness of the environment and the non-colocation of sensors and actuators, was shown to be the most fundamental, since it is of concern for even a single jointed robot. Kinematic stability, which relates to the destabilizing interaction of the inertia and the J^{-1} matrix, was shown to be the bane of hybrid control as it is popularly known [Raibert and Craig, 1981]. Note that this issue arises for multiple joints only, but for as few as two parallel joints. Finally, kinestatic stability, which refers to the instability associated with the decomposition of force and motion feedback signals in order to generate consistent commands, requires at least three joints to occur, and is primarily a problem for higher level constraint formulation.

The chapter ended with a brief discussion of the dc servomotor, which is the type of actuator commonly used on small to medium industrial robots, like the GE P60. Contrary to popular belief, a voltage-controlled dc servomotor, as described in this chapter, can be readily adapted to perform hybrid twist and wrench control tasks. In fact, as demonstrated in Chapter 6, this type of actuator is ideal for implementation of the cross-coordinated control strategy developed here.

CHAPTER 3
ANALYSIS OF COMMERCIAL ROBOT SYSTEM

3.1 Introduction and Objective

In the previous chapter, fundamental concepts essential to the development of the cross-coordinated control strategy were introduced. This chapter continues the developmental process by outlining the kinematic, dynamic, and control system analyses necessary to actually implement this strategy on the GE P60.

Until quite recently, the field of kinematics was focused on single-degree-of-freedom devices for the transmission, control, or constraint of relative motion between two rigid bodies. Since most practical mechanisms were planar, the methods of analysis were largely limited to the two dimensional problem. *This is all changing, mainly* due to the introduction of robots. These are typically spatial mechanisms, and the time honored, planar methods of analysis and synthesis are no longer adequate to handle this new challenge.

Coordinate transformations for spatial linkages, presented over two decades ago [Hartenberg and Denavit, 1964], have found direct application to robot manipulators. The use of these methods for the analysis of displacements and mechanics is presented in Paul [1981], Craig [1986], Asada and Slotine [1987], Fu et al. [1987] and others.

However, when Paul [1981] used these transforms to develop the six-by-six Jacobian matrix relating joint displacements to rotations and translations of the end-effector, he, like most others, made no mention of the fact that the six columns of this

matrix are precisely the six screw coordinates of the joints. Indeed, we often hear about the need to express things "canonically" as a means toward the simplest possible analytical formulation. It was Hunt [1984] who pointed out that the instantaneous screw is, of itself, canonical! Furthermore, the screw "fuses important parts of statics and kinematics into an important and beautiful union" [Hunt, 1984, page 262], kinestatics.

An alternative method for displacement analysis [Duffy, 1980] was used for this work, because its geometical formulation lends itself more readily to the use of screw coordinates. This methodology, which makes use of spherical geometry and the principle of transference, was applied to the GE P60. Kinematics alone, however, were not adequate to develop a control strategy applicable to environmentally constrained tasks, particularly a strategy that is to actually be implemented in the laboratory.

Hence, equally important to this process was the development of realistic models of the plant dynamics. These models must represent the dominant dynamics over the bandwidth planned for the closed-loop system. Failure to accurately represent lightly damped poles, and to compensate for them, creates the risk that the closed-loop system will become unstable, or at least exhibit very poor performance.

The main problem with most published models for industrial robots, as observed by Good et al. [1985], was that they assume the dynamic behavior is adequately represented by interconnected rigid bodies, driven by actuators which are either pure torque sources, or first order lags. There are very few published experimental results to support this assumption.

In contrast, this research looks at the significance of the dynamic cross-coupling effects of the link inertias, as well as the dynamics of the actuators themselves, through

both analysis and experimentation. Further, the development of actuator transfer functions is done in the discrete domain from the outset, rather than the conventional approach of developing continuous time models [Good et. al, 1984], which must then be converted to the discrete domain for digital control.

The final section of this chapter concerns itself with the motion control system provided with the industrial robot by the vendor. This analysis is important to determine the best way to integrate the supplemental wrench control architecture with the existing system in order to realize the hybrid twist and wrench controller for cross-coordinated control.

3.2 Kinematics

The solution of the "inverse kinematics" problem is key to any advanced control strategy. *Inverse kinematics means the determination* of the joint positions that place the end-effector into a specified position and orientation. The reason this is so important is that the Jacobian matrix is a function of the values of those joint displacements; and it is this Jacobian, the columns of which being the screw coordinates of the joints, which shows the connection between static and instantaneous kinematic constraints.

Using screw coordinates [Hunt, 1978], the instantaneous end-effector motion for a serial, six degree-of-freedom manipulator with revolute joints can be written as

$$\begin{bmatrix} \omega \\ v \end{bmatrix} = J \begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \\ \omega_4 \\ \omega_5 \\ \omega_6 \end{bmatrix} \tag{3.1}$$

where ω_1, ω_2, ..., ω_6 are the six actuator velocities, ω is the end-effector angular velocity, v is the translational velocity of a point in the end-effector coincident with the origin and J is the six-by-six Jacobian matrix defined as

$$J \equiv \begin{bmatrix} S_1 & S_2 & S_3 & S_4 & S_5 & S_6 \\ r_1 \times S_1 & r_2 \times S_2 & r_2 \times S_3 & r_4 \times S_4 & r_5 \times S_5 & r_6 \times S_6 \end{bmatrix} \tag{3.2}$$

where each column is the screw coordinates of the corresponding rotary joint. These also happen to be scalar multiples of lines, as discussed in Chapter 2. Besides mapping the instantaneous kinematics between Cartesian and actuator space, the Jacobian matrix also plays an important role in solving the statics problem.

In static equilibrium, *the joint torques must exactly* balance the wrench applied at the end-effector (as well as the other loads). Ignoring those other loads for the moment, it is possible to develop the relationship between that wrench and the required joint torques. Applying the principle of virtual work for infinitesimal displacements allows us to make certain observations about the static case. Since work, which has the *units* of energy, is a scalar quantity its value must be the same, regardless of the coordinates used to represent the problem. Thus, neglecting losses, the work done in Cartesian space must equal the work done in joint space.

Consider the virtual (infinitesimal) displacements of the joints to be $\delta\theta_i$, and the virtual displacements of the end-effector to be δp_i. By convention, these virtual displacements must conform to geometric constraints, represented here by the Jacobian matrix, though not necessarily to the other laws of motion. Let the work done by the joints be

$$\delta(\text{Work})_1 = \tau_i{}^T \delta\theta_i \tag{3.3}$$

where τ_i are the joint torques and the work done by the wrench, w, be

$$\delta(\text{Work})_2 = w^T \delta p_i \tag{3.4}$$

According to the principle of virtual work, the robot arm will be in equilibrium only if the virtual work determined by

$$\delta(\text{Work}) = \delta(\text{Work})_1 - \delta(\text{Work})_2 \tag{3.5}$$

vanishes for an arbitrary virtual displacement.

For an infinitesimal displacement (due to an infinitesimal time step), $\delta p_i = J\delta\theta_i$, from Equation 3.1. Making this substitution

$$\delta(\text{Work}) = (\tau^T - w^T J)\, \delta\theta_i \tag{3.6}$$

or

$$\delta(\text{Work}) = (\tau - J^T w)^T \delta\theta_i \tag{3.7}$$

In order to satisfy the principle of virtual work, Equation 3.7 must vanish for any virtual displacement $\delta\theta_i$, thus

$$\tau = J^T w \tag{3.8}$$

This equation maps the end-effector wrench directly into joint torques, thus obviating the need for a dynamic model to be explicitly included in the force control loop. However, a dynamic model is used implicitly, since the empirically derived tranfer function will govern the servo compensator design, as discussed later in the chapter. Joint torques, however, may not necessarily be the most convenient way to formulate these torque commands.

Modern industrial robots have highly geared transmissions. Thus, the incremental motion of the joints, $\delta\theta_i$, is related to the incremental motion of the motors, $\delta\theta_{mi}$, by a matrix G such that

$$\delta\theta_i = G\,\delta\theta_{mi} \tag{3.9}$$

For a manipulator with a purely serial kinematic structure, this G matrix is diagonal. However, for those robots that include a parallel drive mechanism within the structure, such as the GE P60, off-diagonal terms may be present.

Using the convention that the subscripts 1, 2, ..., 6 refer to the swing, horizontal, vertical, bend, roll and twist axes, respectively. The G matrix for the GE P60 is of the form

$$\begin{bmatrix} g_{11} & 0 & 0 & 0 & 0 & 0 \\ 0 & g_{22} & 0 & 0 & 0 & 0 \\ 0 & g_{32} & g_{33} & 0 & 0 & 0 \\ 0 & 0 & 0 & g_{44} & 0 & 0 \\ 0 & 0 & 0 & 0 & g_{55} & 0 \\ 0 & 0 & 0 & 0 & 0 & g_{66} \end{bmatrix}$$

where

$$g_{22} = -g_{32} \tag{3.10}$$

This is fairly typical of those industrial robots which make use of "parallel structures", such as the GE P60 [Rivin, 1988]. All of the elements of this matrix can be treated as constants except g_{22} $(-g_{32})$ and g_{33}. The values of these elements are configuration dependent, since the horizontal and vertical joint axes are driven by a five-bar mechanism. These functions, like the constant values for the other elements, were determined empirically.

The equations already developed for joint displacements can still be used for the actuator displacements if one defines a modified Jacobian matrix, J^*, such that

$$J^* = J\,G \tag{3.11}$$

where J is the Jacobian matrix for serial manipulators and G is the matrix defined earlier. Using this modified Jacobian matrix, Equation 3.8 now maps the end-effector wrench into a vector of motor torques, instead of joint torques.

One advantage of this formulation is that the rotor dynamics are essentially configuration invariant. Hence, by referring the wrench to the motor, the problem is simplified conceptually. This works well in practice since the rotor inertias tend to dominate the link inertias for highly geared, industrial robots, such as the GE P60.

Consequently, the Jacobian matrix, which is the matrix of screws describing the geometric constraints provided by the joints, is the principal transformation between the joints and the end-effector, both for static constraints (actually the Jacobian transpose)

as well as for instanteous kinematics. However, as already stated, this matrix, as well as the G matrix, are dependent on the joint positions. To solve for them, given that we know the desired position and orientation of the robot end-effector, requires that we solve the inverse kinematics. The procedure used is called the reverse analysis.

3.2.1 Reverse Analysis

The notation used throughout this analysis is that presented by Duffy [1980]. A manipulator is considered to be constructed of a series of rigid links as shown in Figure 3.1. Observe that a link connects the two kinematic pair (joint) axes $\underline{S}_i$ and $\underline{S}_j$. The perpendicular distance between the joints is a_{ij}, with the vector along this mutual perpendicular labelled $\underline{a}_{ij}$. The twist angle between the joints is labelled α_{ij} and is measured in a right handed sense about $\underline{a}_{ij}$.

For this work, we limit ourselves to *considering the revolute joint*, as shown in Figure 3.2, as it is the only type of joint axis present on the GE P60. The perpendicular distance between the links is labelled S_{jj}, and is termed the offset. The relative angle between the two links is shown as θ_j, the joint displacement, and is measured in a right handed sense about the vector $\underline{S}_j$.

Hence, there are four parameters that describe the geometry of the manipulator. These include the joint displacement (θ_j), twist angle (α_{ij}), offset (S_{jj}) and link length (a_{ij}). Note that only the joint displacements, θ_j, are unknown; and for all revolute joints, the joint displacements are joint angles. The twist angles, offsets, and link lengths are known constants. The values for the sine and cosine for the twist and joint angles can be determined from the following equations, respectively,

$$s_{ij} = | \underline{S}_i \; \underline{S}_j \; \underline{a}_{ij} | \tag{3.12}$$

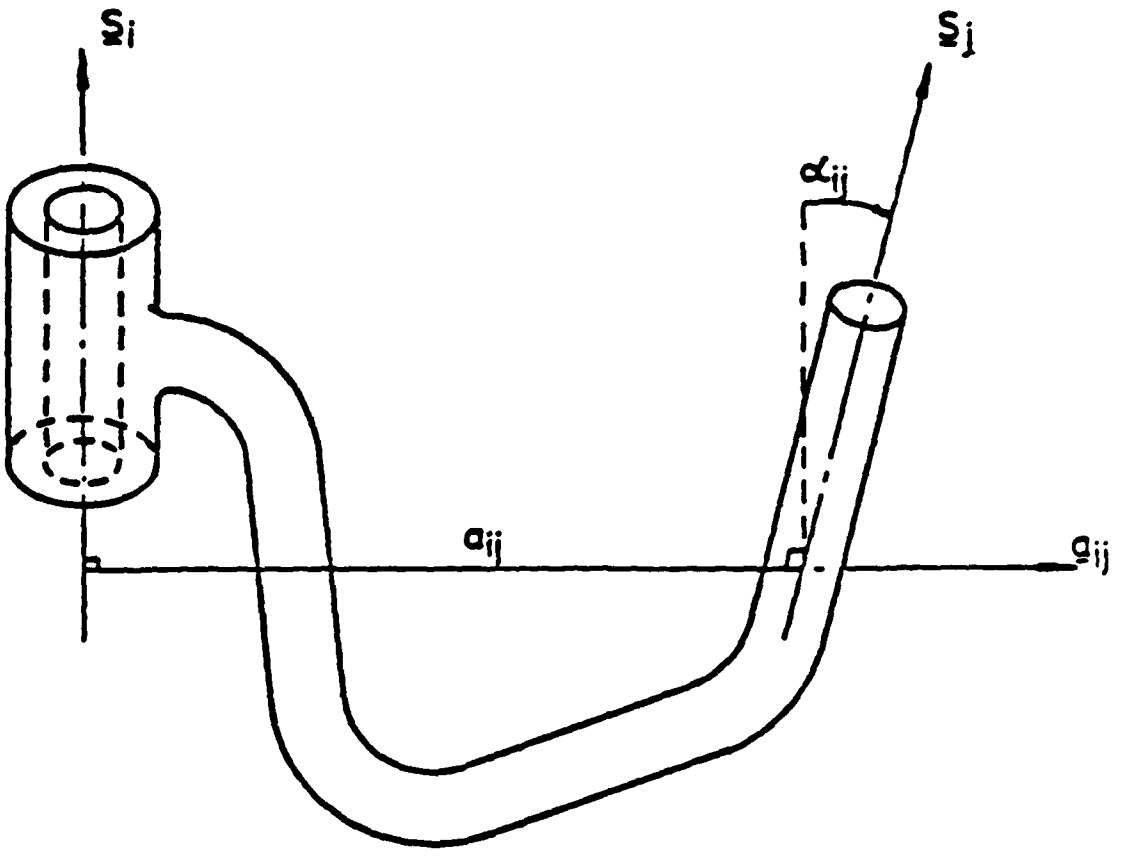


Figure 3.1

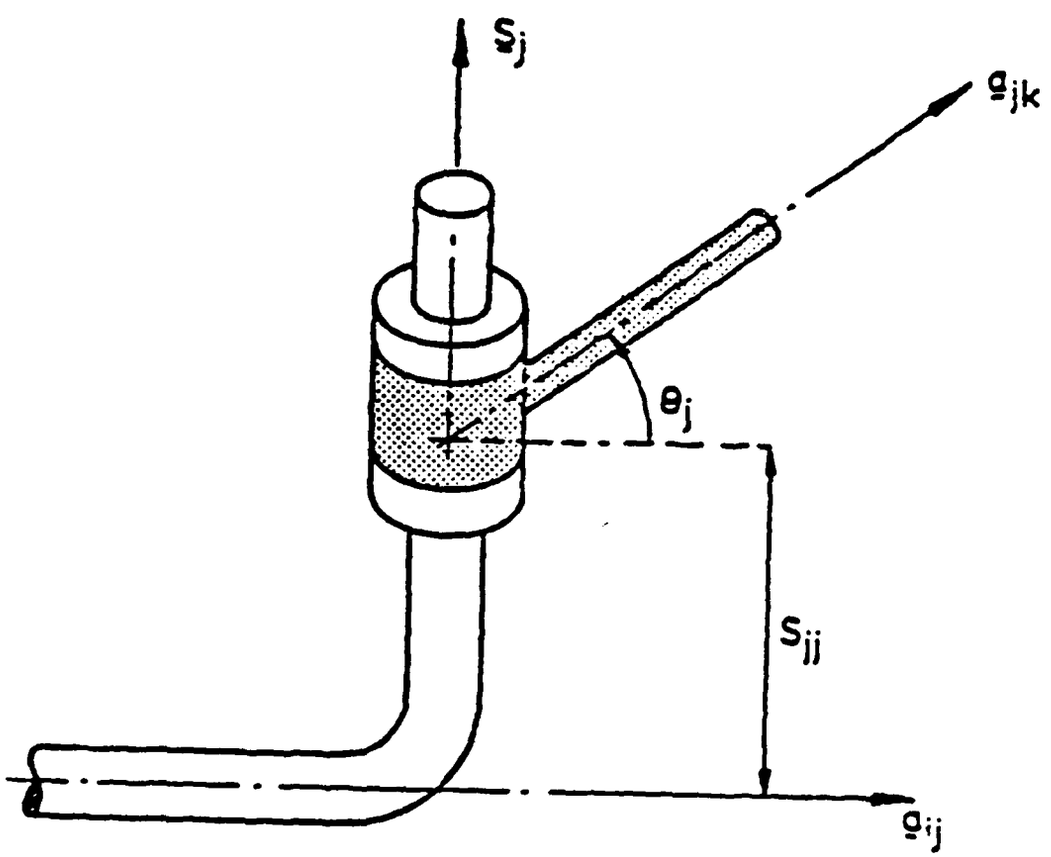


Figure 3.2

$$c_{ij} = \underline{S}_i \cdot \underline{S}_j \tag{3.13}$$

$$s_j = | \underline{a}_{ij} \, \underline{a}_{jk} \, \underline{S}_j | \tag{3.14}$$

$$c_j = \underline{a}_{ij} \cdot \underline{a}_{jk} \tag{3.15}$$

where $s_{ij} = \sin \alpha_{ij}$ and $s_j = \sin \theta_j$. The determinant notation is used to denote the scalar triple product.

$$| \underline{a} \, \underline{b} \, \underline{c} | \equiv \underline{a} \times \underline{b} \cdot \underline{c} \tag{3.16}$$

Shown in Figure 3.3 is a sketch of the GE P60 robot. A skeletal representation of the P60 is shown is Figure 3.4. Particularly important are the three parallel axes: horizontal, vertical and bend, (or upper arm, forearm and pitch) which are the key feature for the reverse analysis of this manipulator. The joint axes are labelled sequentially with unit vectors, $\underline{S}_i$ (i=1, 2, ..., 6). The directions of the common normal between two successive joint axes S_i and S_j are labelled with the unit vectors $\underline{a}_{ij}$ (ij=12, 23, ..., 67). Note that for clarity, not all of these vectors are shown.

As already noted, the link lengths (a_{ij}), offsets (S_{jj}) and twist angles (α_{ij}) are constants, specific to the geometry of manipulator. For the GE P60, these constants are listed in Table 3.1. In addition to these, S_{66} and a_{67} are selected such that the end point of the vector $\underline{a}_{67}$ is the point of interest of the tool, such as the tip of a welding rod. Once a tool has been specified, constants for S_{66} and a_{67} are known.

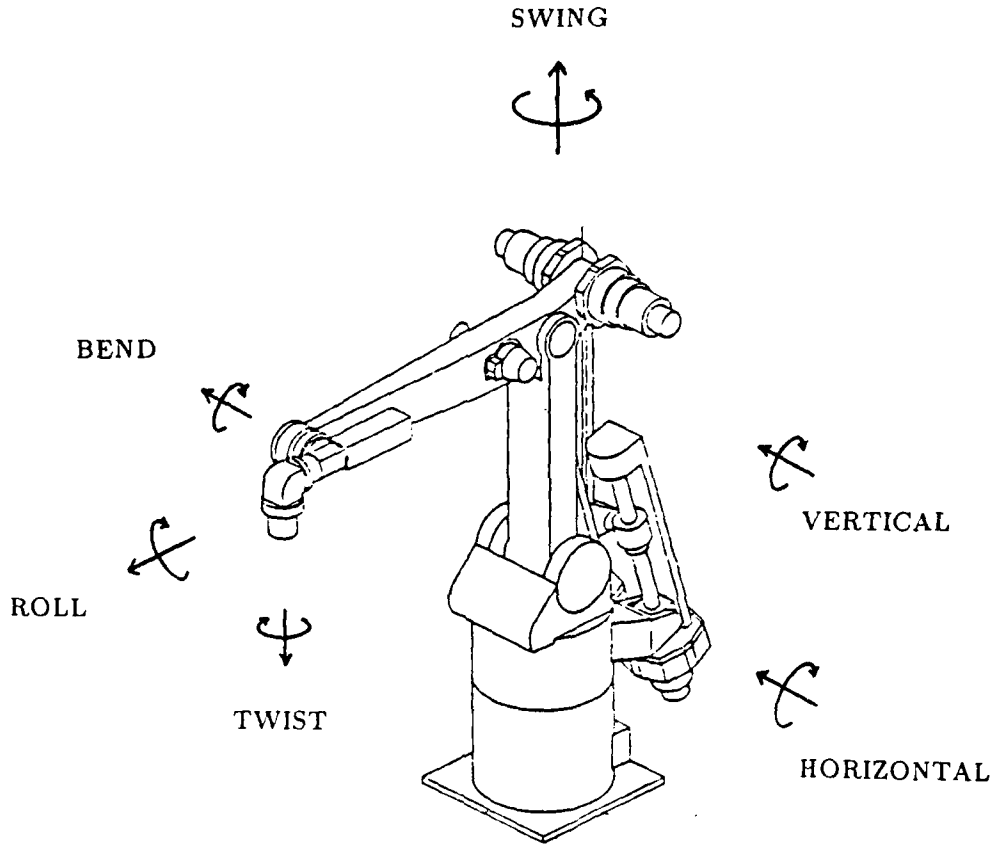


Figure 3.3 General Electric P60 Industrial Robot

SKELETAL MODEL OF THE GE P60

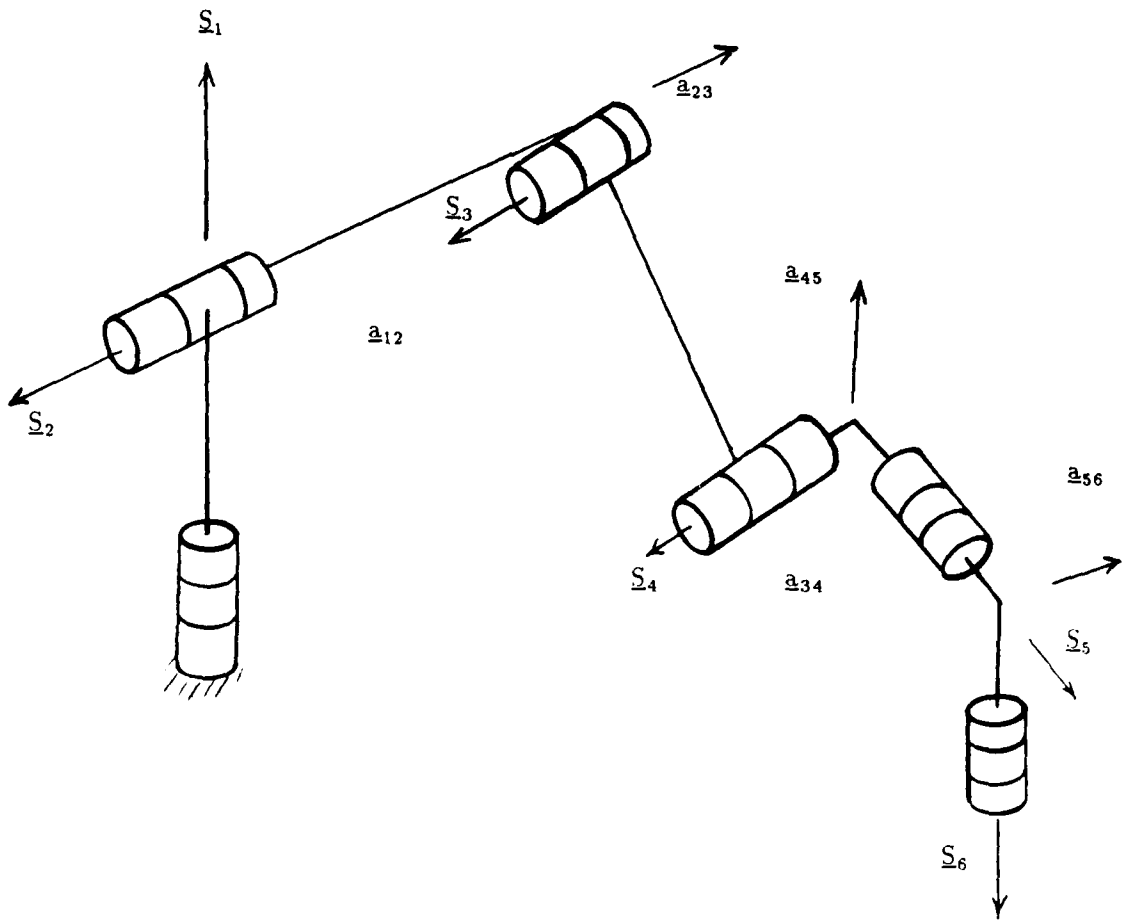


Figure 3.4

Table 3.1 GE P60 constants.

$S_{11} = *$	$a_{12} = 0$	$\alpha_{12} = 90^{\circ}$
$S_{22} = 0$	$a_{23} = 70\text{cm}$	$\alpha_{23} = 0^{\circ}$
$S_{33} = 0$	$a_{34} = 90\text{cm}$	$\alpha_{34} = 0^{\circ}$
$S_{44} = 9.8\text{cm}$	$a_{45} = 0$	$\alpha_{45} = 90^{\circ}$
$S_{55} = 14.5\text{cm}$	$a_{56} = 0$	$\alpha_{56} = 90^{\circ}$

* to be determined from closing the loop

Although the link lengths a_{12}, a_{45} and a_{56} are zero, it is still necessary to specify the direction of their associated unit vectors in order to have the sense of the axis about which to measure the corresponding twist angle. The vector $\underline{a}_{ij}$ must be perpendicular to the plane specified by the vectors $\underline{S}_i$ and $\underline{S}_j$ and, as such, can have two possible directions. The direction is chosen as that parallel to $\underline{S}_i \times \underline{S}_j$, and the twist angles listed were determined by this convention.

The reverse analysis of the manipulator consists of determining the values of the joint angles necessary to place the tool in the desired position and orientation. It is complicated by the fact that there are most often more than one set of joint angles which will satisfy the specification. However, the reverse analysis approach to the inverse kinematics problem has the advantage that all sets are determined, as opposed to numerical iteration techniques which find only one set.

It turns out that because of the geometry of the GE P60, namely the three parallel joint axes, there are eight possible solution sets, ignoring joint limitations.

However, when these joint limitations are included, the number reduces considerably such that, for most of the work space, there is only two solution sets possible. The limits for the rotation of these angles in absolute coordinates are shown in Table 3.2.

The first step in this analysis is to establish a fixed coordinate system, which is shown in Figure 3.4. The origin is located at the intersection of vectors $\underline{S}_1$ and $\underline{S}_2$. The Z axis is along $\underline{S}_1$ and the X axis bisects the allowable range of rotation of the swing axis.

Using this fixed coordinate system, the location of the tool tip is specified as $\underline{R}$, while the direction cosines of the vectors $\underline{S}_6$ and $\underline{a}_{67}$ complete the specification of the orientation. Although these three vectors have nine components, the last two related by

$$\underline{S}_6 \cdot \underline{S}_6 = 1 \tag{3.17}$$

$$\underline{a}_{67} \cdot \underline{a}_{67} = 1 \tag{3.18}$$

$$\underline{S}_6 \cdot \underline{a}_{67} = 0 \tag{3.19}$$

so that the three vectors actually represent only the six independent parameters necessary to completely specify the position and orientation of a rigid body in space.

Adopting this procedure, the end-effector is connected to ground by a hypothetical link in a process called "closing the loop" [Lipkin and Duffy, 1985b]. Hence, the problem of determining the sets of joint displacements satisfying the reverse analysis is reduced to analyzing an equivalent spatial mechanism of mobility one.

Table 3.2 Joint Limits

Axis	Joint Motion Range [degrees]
Base (Swing)	± 150
Upper Arm (Horizontal)	+ 46 − 40
Forearm (Vertical)	+ 20 (+ 90)* − 42 (+ 90)
Pitch (Bend)	± 110
Roll	± 180
Twist	± 300

* The range of joint motion is given in absolute angles. For relative angles, add the value in parentheses.

Closing the loop consists of determining the five constraint parameters S_{77}, a_{71}, S_{11}, α_{71} and $(\theta_1-\phi_1)$, together with the input angle of the spatial mechanism, θ_7. Angle ϕ_1 is the first manipulator joint angle measured from the fixed X axis to $\underline{a}_{12}$, measured about $\underline{S}_1$. It is determined by subtracting $(\theta_1-\phi_1)$ from θ_1. While the procedure is not trivial, it is identical for all six axis, serial manipulators, and therefore is not included here. The interested reader should refer to Lipkin and Duffy [1985b] for a detailed development and sample application of this procedure.

The process continues with the writing of the vector loop equation for the closed loop mechanism

$$S_{11}\underline{S}_1 + a_{12}\underline{a}_{12} + S_{22}\underline{S}_2 + a_{23}\underline{a}_{23} + S_{33}\underline{S}_3 + a_{34}\underline{a}_{34} - S_{44}\underline{S}_4 + a_{45}\underline{a}_{45} + S_{55}\underline{S}_5 + a_{56}\underline{a}_{56} + S_{66}\underline{S}_6 + a_{67}\underline{a}_{67} + S_{77}\underline{S}_7 + a_{71}\underline{a}_{71} = 0 \qquad (3.20)$$

which reduces to

$$S_{11}\underline{S}_1 + a_{23}\underline{a}_{23} + a_{34}\underline{a}_{34} - S_{44}\underline{S}_4 + S_{55}\underline{S}_5 + S_{66}\underline{S}_6 + a_{67}\underline{a}_{67} + S_{77}\underline{S}_7 + a_{71}\underline{a}_{71} = 0 \qquad (3.21)$$

due to the zero link and offset parameters already given.

It is most convenient to express this vector equation in terms of three scalar equations, each corresponding to one component. Using set 14 [Duffy, 1980], the vector loop equation can thus be expressed in this fashion, viz.

$$S_{11}\begin{bmatrix} 0 \\ -s_{12} \\ c_{12} \end{bmatrix} + a_{23}\begin{bmatrix} c_2 \\ -s_2 \\ 0 \end{bmatrix} + a_{34}\begin{bmatrix} W_{5671} \\ -U^*_{456712} \\ U_{456712} \end{bmatrix} + S_{44}\begin{bmatrix} X_{5671} \\ Y_{5671} \\ Z_{5671} \end{bmatrix} +$$

$$S_{55}\begin{bmatrix} X_{671} \\ Y_{671} \\ Z_{671} \end{bmatrix} + S_{66}\begin{bmatrix} X_{71} \\ Y_{71} \\ Z_{71} \end{bmatrix} + a_{67}\begin{bmatrix} W_{71} \\ -U^*_{712} \\ U_{712} \end{bmatrix} + S_{77}\begin{bmatrix} X_1 \\ Y_1 \\ Z_1 \end{bmatrix} +$$

$$a_{71}\begin{bmatrix} W_1 \\ U^*_{12} \\ U_{12} \end{bmatrix} = \underline{0} \tag{3.22}$$

Substituting known parameters into the Z component equation yields

$$Ac_1 + Bs_1 + D = 0 \tag{3.23}$$

where

$$A = a_{67}s_7c_{71} - S_{77}s_{71} - c_{71}c_7S_{66}$$

$$B = S_{66}s_7 + a_{67}c_7 + a_{71}$$

$$D = -S_{44}$$

Making the substitutions

$$c_1 = \frac{1-x_1^2}{1+x_1^2}, \quad s_1 = \frac{2x_1}{1+x_1^2}, \quad x_1 \equiv \tan\frac{\theta_1}{2} \tag{3.24}$$

in Equation 3.23 and solving for x_1 yields

$$x_1 = \frac{-B \pm \sqrt{A^2 + b^2 - D^2}}{D - A} \tag{3.25}$$

The two values for θ_1 result from the $\pm$ sign in front of the radical. Each value can then be added to $(\phi_1 - \theta_1)$ to obtain corresponding values for ϕ_1.

From the subsidiary sine law for a spherical hexagon

$$X_{17} = X_{3456} \tag{3.26}$$

and

$$Y_{17} = -X^*_{3456} \tag{3.27}$$

This is used since a spherical hexagon reduces to a spherical pentagon if three $\underline{S}_i$ vectors are parallel, as was the case here. Thus, since $\underline{S}_2$, $\underline{S}_3$ and $\underline{S}_4$ are parallel, Equation 3.26 reduces to

$$X_{17} = X_{56} \tag{3.28}$$

and

$$Y_{17} = X^*_{56} \tag{3.30}$$

Similarly

$$Z_{71} = \overline{Z}_5 \tag{3.31}$$

This expression reduces to

$$c_5 = c_{7_i}c_7c_1 - s_7s_1 \tag{3.32}$$

Solving this expression results in two values for θ_5 for each combination of θ_7 and θ_1, since only the cosine of θ_5 is available.

Equation 3.26 and Equation 3.27 can now be expanded to yield

$$s_6 = \frac{-s_{71}c_1}{s_5} \tag{3.33}$$

and

$$c_6 = \frac{s_1c_7 + c_{71}c_1s_7}{s_5} \tag{3.34}$$

From these two expressions, a unique θ_6 can be found for any set of values of $\theta_{\overline{i}}$, θ_1 and θ_5.

Expanding the X and Y components of the vector loop equation yields

$$a_1c_2 + b_1c_{2+3} + d_1 = 0 \tag{3.35}$$

and

$$a_2 s_2 + b_2 s_{2+3} - d_2 = 0 \tag{3.36}$$

Squaring and adding these expressions results in

$$c_3 = \frac{d_1^2 + d_2^2 - a_1^2 - b_1^2}{2\, a_1 b_1} \tag{3.37}$$

where

$$a_1 = a_{23}\,, \quad b_1 = a_{34}$$

$$a_2 = a_{23}\,, \quad b_2 = a_{34}$$

$$d_1 = S_{55}(s_6 c_7 c_1 - s_6 s_7 c_{71} s_1 - s_{71} c_6 s_1) + S_{66}(s_7 c_1 + c_{71} c_7 s_1) +$$
$$a_{67}(c_1 c_7 - s_1 s_7 c_{71}) + S_{77} s_{71} s_1 + a_{71} c_1$$

$$d_2 = S_{55}(c_6 c_{71} - s_6 s_7 s_{71}) + S_{66} s_{71} c_7 - a_{67} s_7 s_{71} - S_{77} c_{71} - S_{11}$$

This expression yields two θ_3 values for each set of θ_7, θ_1, θ_5 and θ_6.

Regrouping the X and Y component expression assuming θ_3 is known yields

$$\begin{bmatrix} a_3 & b_3 \\ a_4 & b_4 \end{bmatrix} \begin{bmatrix} c_2 \\ s_2 \end{bmatrix} = \begin{bmatrix} d_3 \\ d_4 \end{bmatrix} \tag{3.38}$$

where

$$a_3 = a_{23} + a_{34}c_3 \qquad a_4 = a_{34}s_3$$
$$b_3 = -a_{34}s_3 \qquad b_4 = a_{23} + a_{34}c_3$$
$$d_3 = -d_1 \qquad d_4 = -d_2$$

Using Cramer's rule we have

$$c_2 = \frac{d_3b_4 - d_4b_3}{a_3b_4 - a_4b_3} \tag{3.39}$$

$$s_2 = \frac{a_3d_4 - a_4d_3}{a_3b_4 - a_4b_3} \tag{3.40}$$

These two expressions yield a unique θ_2 for each set of θ_7, θ_1, θ_5, θ_6 and θ_3.

Finally,

$$X_{67123} = s_{45}s_4 \tag{3.41}$$

and

$$Y_{67123} = s_{45}c_4 \tag{3.42}$$

yield a unique value for θ_4. The eight sets of joint angles thus obtained, without regard to joint limitations, are depicted in Figure 3.5.

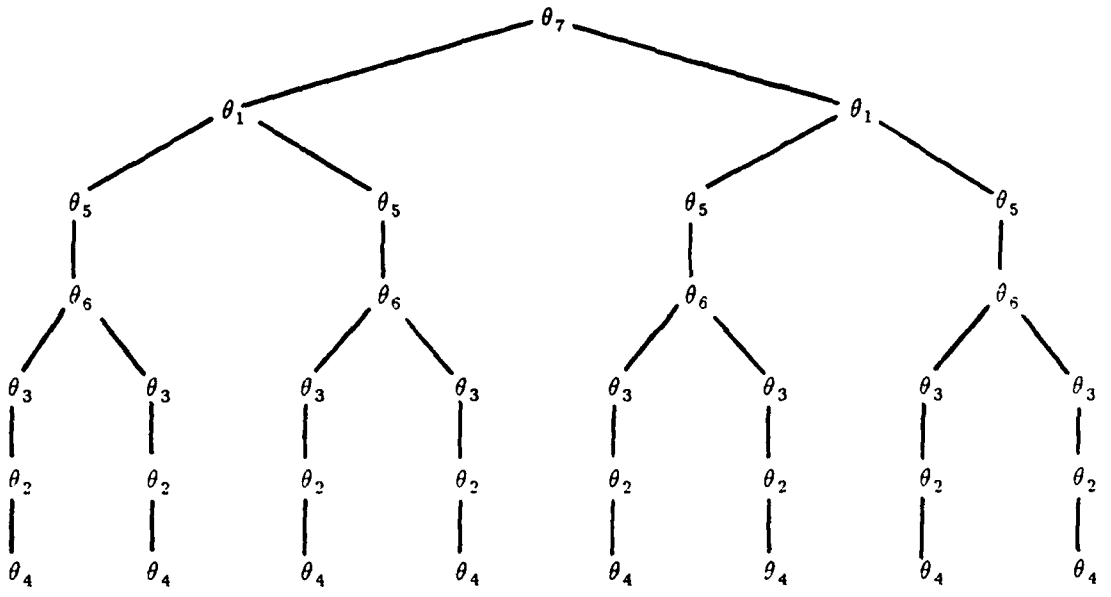


Figure 3.5 Possible Sets of Joint Angles

3.2.2 Forward Analysis

For the forward displacement analysis, it is assumed that the joint angles are known. Using them, one can uniquely determine the position and orientation of the end-effector, and thus verify the results obtained from the reverse analysis.

The direction cosines of $\underline{S}_6$ and $\underline{a}_{67}$ in the coordinate system located in the hand, namely the sixth coordinate system, are simply (0,0,1) and (1,0,0) by definition. They may be expressed in the first coordinate system by five successive applications of the rotation matrix

$$A_{ji} = \begin{bmatrix} c_j & -s_j & 0 \\ s_j c_{ij} & c_j c_{ij} & -s_{ij} \\ s_j s_{ij} & c_j s_{ij} & c_{ij} \end{bmatrix} \tag{3.43}$$

Thus we can express vectors from the 6th coordinate system in terms of the 1st coordinate system by using

$$\begin{bmatrix} x^1 \\ y^1 \\ z^1 \end{bmatrix} = A_{21}\, A_{32}\, A_{43}\, A_{54}\, A_{65} \begin{bmatrix} x^6 \\ y^6 \\ z^6 \end{bmatrix} \tag{3.44}$$

These direction cosines can also be expressed in the fixed coordinate system using

$$\begin{bmatrix} x^f \\ y^f \\ z^f \end{bmatrix} = M \begin{bmatrix} x^1 \\ y^1 \\ z^1 \end{bmatrix} \tag{3.45}$$

where

$$M = \begin{bmatrix} \cos\phi_1 & -\sin\phi_1 & 0 \\ \sin\phi_1 & \cos\phi_1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{3.46}$$

Combining these expressions one obtains

$$\underline{S}_6^f = M\, A_{21}\, A_{32}\, A_{43}\, A_{54}\, A_{65}\, \underline{S}_6^6 \tag{3.47}$$

and

$$\underline{a}_{67}^f = M\, A_{21}\, A_{32}\, A_{43}\, A_{54}\, A_{65}\, \underline{a}_{67}^6 \tag{3.48}$$

The last parameter to be determined is the position vector of the point of interest. The vector loop equation can be written as

$$a_{23}\underline{a}_{23} + a_{34}\underline{a}_{34} - S_{44}\underline{S}_4 + S_{55}\underline{S}_5 + S_{66}\underline{S}_6 + a_{67}\underline{a}_{67} = \underline{R}^i \tag{3.49}$$

Using the direction cosines from set one results in the components of $\underline{R}^1$. These values must then be rotated into the fixed frame using

$$\underline{R}^f = M\,\underline{R}^1 \tag{3.50}$$

The forward analysis is now complete.

3.3 Dynamics

Nearly all models for robot dynamics presented in the literature are based on the assumption that the arm consists of a series of rigid bodies connected by kinematic pairs. With this model, the dynamics reduces to the following familiar equation.

$$\tau = H(\theta)\ddot{\theta} + C(\theta,\dot{\theta}) + G(\theta) + J(\theta)^T w \tag{3.51}$$

Here θ is a vector of joint angles, H is the inertia matrix, C is a vector representing centrifugal and Coriolis effects, G is a vector of gravity effects, J^T is the Jacobian transpose matrix relating the vector of forces and moments at the end-effector (wrench), w, to the joint torques, τ, applied by the actuators, namely the Jacobian matrix discussed earlier.

This equation, as applied to serial manipulators, has been exhaustively discussed with special emphasis on such things as inertia effects, configuration dependence, payload dependence, cross-coupling effects and non-linearity due to geometry. Usually, the actuators are modelled as pure torque sources, or as first order lags. Not surprisingly, as observed by Good et al. [1985], very little experimental evidence to support such modelling is found in the literature.

Consider the following representation of this equation, for no applied wrench, as presented by Asada and Youcef-Toumi [1987].

$$\tau_i = H_{ii}\ddot{\theta}_i + \sum_{j \neq i} H_{ij}\ddot{\theta}_j + \sum_j \sum_k \left(\frac{\partial H_{ij}}{\partial \theta_k} - \frac{1}{2} \frac{\partial H_{jk}}{\partial \theta_i} \right) \dot{\theta}_j \dot{\theta}_k + \tau_{gi} \qquad (3.52)$$

Here the first term is the diagonal elements while the second term is the off-diagonal elements of the inertia matrix. The third term represents non-linear velocity torques resulting from configuration dependence. For a model with decoupled inertia, the second term is zero, while for a configuration invariant case, the third term is zero. For both a decoupled and configuration invariant case, Equation 3.52 reduces to

$$\tau_i = H_{ii}\,\ddot{\theta}_i + \tau_{gi} \qquad (3.53)$$

An inertia decoupled and configuration invariant designed direct-drive robot manipulator employing a "special five-bar-link parallel drive mechanism" was reported by Asada and Youcef-Toumi [1987, page 66]. In point of fact, this mechanism bears a striking resemblance, kinematically, to the parallel drive mechanisms commonly found on many industrial robots, such as the GE P60. Accepting the significance of their

design for direct-drive manipulators, even Asada and *Youcef-Toumi* [1987] admit the relative insignificance of cross-coupled and changing link inertias when torques are referred to the actuator of a typical industrial robot.

Referring the torque equation to the motor, the dynamic equation can now be written [Asada and Youcef-Toumi, 1987]

$$\tau_{mot} = (J_{rot} + \frac{J_{arm}}{n_i^2})\alpha_i + \frac{\tau_{coup}}{n_i} + \frac{\tau_{load}}{n_i} \tag{3.54}$$

where τ_{mot} is the motor torque, τ_{coup} the coupling torque, τ_{load} the load torque, J_{rot} the rotor inertia of the motor, J_{arm} the arm link inertias, α_i the motor accelerations and n_i the respective gear ratios. Recognizing that robot transmissions, such as harmonic drives, have gear ratios on the order of 100:1 or more, it's clear that the link inertias are reduced by a ratio of at least 10000:1, while the coupling and load torques reduced by 100:1. Furthermore, these servomotors typically use velocity feedback, which also tends to make them insensitive to load changes. Consequently, the utility of this type of analysis for industrial robot control is questionable.

Advocates of the direct-drive architecture counter that transmissions generate backlash [Asada and Youcef-Toumi, 1984 and 1987], a highly non-linear condition where one input can have two possible output values. Others have said that commercial robots are not suitable for more advanced control strategies, expressly because of these high gear ratios and joint friction [An et al., 1988].

The fact is, mechanical transmissions have been used on commercial robots because required joint velocities are typically small, while joint torque requirements are not. Since the dimensions of an electric motor are determined by the torque, a high

speed, low torque motor, even with a transmission, is smaller, cheaper, and has a smaller rotor inertia than the low speed, high torque motors required for direct drive. Also, modern mechanical transmissions are extremely effective at converting speed to torque with high efficiency. Furthermore, recent experimental work [Youcef-Toumi and Ro, 1986] has revealed backlash-like electrical effects in these high torque motors. Other problems such as high cost, high motor weight, energy dissipation difficulties, sensitivity to inertia and load changes as well as duty cycle requirements are causing some advocates of direct-drive technolgy to suggest the use of low reduction gears in their systems [Youcef-Toumi and Nagano, 1987]. Cynics might suggest that these researchers may one day rediscover the transmission.

Table 3.3 shows the results of an experiment performed in order to evaluate the inertia cross-coupling. This experiment was very similiar to that performed by Asada and Youcef-Toumi [1987], which used peak-to-peak ratios of position signals as a measure of the coupling of the arm dynamics. The magnitude ratios reported in Table 3.3 are expressed in Db's, so as to compare them to the results reported by Asada and Youcef-Toumi [1987]. Their results indicated coupling of less that -30 Db for all configurations, which lead them to pronounce their design "well decoupled" [Asada and Youcef-Toumi, 1987, page 149]. Interestingly, the GE P60 showed an even smaller cross-coupling, less than -40 Db, just as it came from the crate.

Do these results imply that this commercial robot is a linear system? Hardly! The important non-linearities, however, have little to do with Equation 3.51. These non-linearities include Coulomb friction, transmission dynamics and current limiters on motor control loops [Sweet and Good, 1984]. Backlash effects, though present, were found to be relatively small, as is typical of modern transmissions such as harmonic

Table 3.3 Link Inertia Coupling Test Results

Arm Configuration ($\alpha_j - \alpha_i$ Degrees)		Oscillation Amplitude Ratio (Db) of Passive Axis to Active Axis	
	[degrees]	0.5 Hz	5 Hz
Effect of forearm motor on upper arm motor	95	−70 [−48]	−42 [−31]
	110	−75 [−57]	−46 [−40]
	130	−75 [−53]	−44 [−39]
Effect of forearm motor on base axis motor	95	−76 [−57]	−50 [−34]
	110	−80 [−59]	−57 [−38]
Effect of upper arm motor on base axis motor	95	−79 [−59]	−64 [−49]

Numbers in brackets are for corresponding tests results on a direct-drive arm reported by Asada and Youcef-Toumi [1987].

drives. Why then this pre-occupation with Equation 3.51? The author can only surmise that since link inertia coupling, Coriolis effects and centrifugal effects are problems that can be quite elegantly quantified, some researchers may prefer to investigate them, rather than tackle the real world problems which are not so conveniently expressed for publication.

The initial dynamic analysis developed here was focussed on the development of continuous time models of the actuators. Bode plots were used. This conventional approach is outlined in Good et al. [1985], and similar results were obtained, viz. the smaller joints out near the wrist showed what appeared to be first order behavior, as shown in Figure 3.6, while the larger joints near the base, which have large masses on either side of the joint, exhibited strong resonance/anti-resonance characteristics, as shown in Figure 3.7. However, this approach suffered from a major drawback, namely the need to take a series of empirically derived transfer functions, combine them, and discretize the result. Furthermore, a model of the end of arm tooling, which had significant dynamics due to the RCC device, would also have to be included.

Some eight ways have been commonly suggested in the literature to discretize a continuous transfer function. A problem arises as to which method should be used. Also, there is a problem in combining these transfer functions of the subsystems which, when taken together, are supposed to describe the overall dynamic system. Åström and Wittenmark [1984] warn that the discretization of the continuous transfer function can, of itself, cause the compensator design to be unstable. Recognizing the need to control the robot with a digital control loop led the author to consider generating a discrete model directly. This approach would obviate the need to choose a discretization approach, and thus avoid the associated uncertainties.

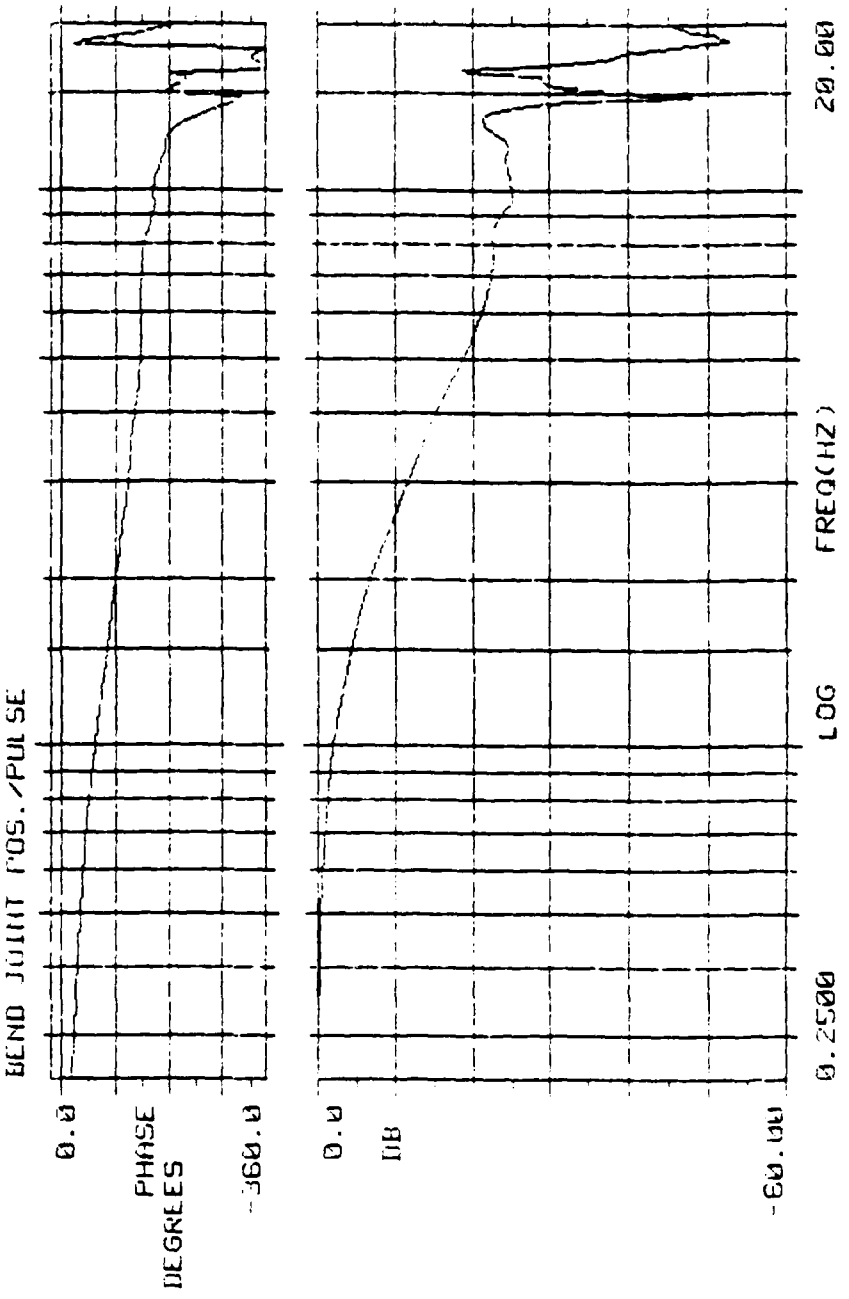

BEND JOINT POS./PULSE
0.0
PHASE
DEGREES
-360.0
0.0
DB
-60.00
0.2500
LOG
FREQ(HZ)
20.00

Figure 3.6

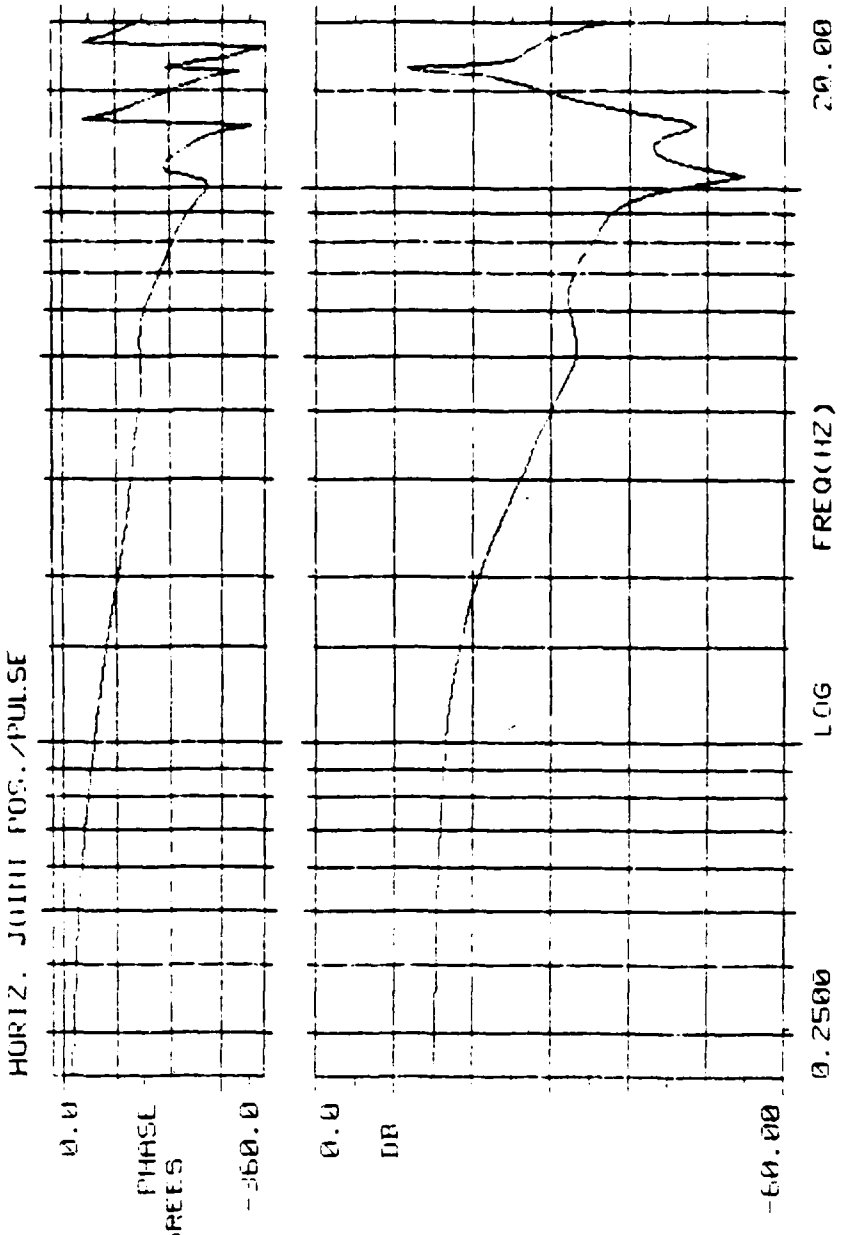

HORIZ. JOINT POS. /PULSE
0.0
PHASE
IN DEGREES
-360.0
0.0
DB
-60.00
0.2500
LOG
FREQ(HZ)
20.00

Figure 3.7

The formulation of a suitable stochastic modelling approach is well presented in a book by Bollinger and Duffie [1988]. While the structure of closed-loop feedback control is very similiar, regardless of whether we are using a continuous or discrete controller, a significant difference arises from the fact that the digital computer operates with specific sets of numbers at discrete intervals. This controller must sample the values of the output process variables at certain instants in time. An error is then calculated, which is used to generate a new input signal. The speed at which the controller can take these samples is called the sample interval, and has a dramatic effect on stability. In fact, the speed of sampling is directly related to how well the process can be controlled.

It is possible to estimate the coefficients of a discrete model for a process by statistical analysis using samples taken from the input and ouput, without precisely knowing the exact nature of the process. For this work a least-squares estimation of the coefficients was used to determine a "best-fit" of the empirical data. This step-response modelling has a very significant advantage over physical modelling in that only the time-domain response of the system is required. This obviates the need for the frequency response test equipment normally employed (and virtually any other test equipment for that matter). The approach is quick, easy to implement and the results explicitly quantifiable as to their statistical significance.

The procedure begins by determining which transfer function is to be generated. Here, a ratio of motor torque units to D/A (digital to analog) input units was used. This was convenient to program. Futhermore, it lent itself quite naturally to the compensator design, as developed in Chapter 4. Having made this decision, the system should then be configured exactly as one intends to operate it. The control computer

generates the input, and also receives and records the response. The transducer to be used for control is the same one that is used for the test. Control computations are made during each execution of the control loop in order to account for computational delay. Finally, the actual tool and workpiece were utilized in the experiment.

Test configuration is significant because this technique will accurately generate the transfer function for a given set of operating conditions, which should obviously be configured as close to the actual operating conditions as possible. With the system at zero input and zero output, an open loop step is applied by the control computer, preferably in the range of values intended for process usage. The output of the transducer to the control computer is then recorded.

There are two important pieces of information to be collected. One is the actual output values, themselves, and the other is the sample rate of the control system. For a design with negligible computational delay, the transducer sample rate provided by the transducer manufacturer should suffice. If this information is not provided, or is suspect, the sample rate can be easily determined experimentally, using an oscilloscope. If the test is performed under a realistic set of operating conditions, as described above, the resulting value should be the same sample rate experienced under closed-loop control.

The discrete model of the process was then fit to the data using an equation of the form [Bollinger and Duffie, 1988]

$$c_n = \alpha_1 c_{n-1} + \alpha_2 c_{n-2} + \cdots + \alpha_r c_{n-r} + \beta_1 m_{n-1-d} + \beta_2 m_{n-2-d} + \cdots + \beta_s m_{n-s-d} + \varepsilon_n \qquad (3.55)$$

where c_n (n=1,2,3,...,N) is the system output, m the system input, N the number of samples, α_i (i = 1,2,3,...,r) the output coefficients, β_j (j=1,2,3,...,s) the input coefficients, d the delay and ε_n the residual error. Equation 3.55 can be expressed in matrix notation by

$$\underline{c} = X\,\hat{\underline{b}} + \underline{\varepsilon} \tag{3.56}$$

where $\hat{\underline{b}}$ is an estimate of the coefficients and

$$X = \begin{bmatrix} c_0 & c_{-1} & \cdots & c_{1-r} & m_{-d} & m_{-1-d} & \cdots & m_{1-s-d} \\ c_1 & c_0 & \cdots & c_{2-r} & m_{1-d} & m_{-d} & \cdots & m_{2-s-d} \\ c_2 & c_1 & \cdots & c_{3-r} & m_{1-d} & m_{1-d} & \cdots & m_{3-s-d} \\ \vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots \\ c_{N-1} & c_{N-2} & \cdots & c_{N-r} & m_{N-1-d} & m_{N-2-d} & \cdots & m_{N-s-d} \end{bmatrix} \tag{3.57}$$

where

$$\underline{c} = \begin{bmatrix} c_1 \\ c_2 \\ c_3 \\ \vdots \\ c_N \end{bmatrix}, \qquad \underline{\varepsilon} = \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \varepsilon_3 \\ \vdots \\ \varepsilon_N \end{bmatrix}, \qquad \underline{b} = \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \vdots \\ \alpha_r \\ \beta_1 \\ \beta_2 \\ \beta_3 \\ \vdots \\ \beta_s \end{bmatrix}$$

The residual errors are

$$\underline{\varepsilon} = \underline{c} - X\,\underline{\hat{b}} \tag{3.58}$$

A positive definite objective function can thus be defined by

$$S(\hat{b}) = \underline{\varepsilon}^T\underline{\varepsilon} \tag{3.59}$$

The corresponding set of normal equations is found by setting

$$\frac{\partial S(\hat{b})}{\partial b_k} = 0 \tag{3.60}$$

for $k = 1, 2, 3, \ldots, r+s$ estimates of the coefficients.

$$\frac{\partial S(\hat{b})}{\partial b_i} = \frac{\partial(\varepsilon^T\varepsilon)}{\partial b_i} = 2\varepsilon^T\frac{\partial \varepsilon}{\partial b_i} \tag{3.61}$$

Using all the $r + s$ equations thus obtained yields

$$X^T[\underline{c} - X\,\underline{\hat{b}}] = \underline{0} \tag{3.62}$$

Solving for $\hat{b}$ yields

$$\underline{\hat{b}} = [X^TX]^{-1}X^T\underline{c} \tag{3.63}$$

The matrix $[X^T X]$, to be inverted, is of order r+s, regardless of the number of observations. Those familiar with APL will immediately recognize the least squares fit described above as a library function called domino, which is coded as

$$b \leftarrow c \boxminus X \tag{3.64}$$

and which obviates the need for any computation at all.

The variance of the estimate is

$$\sigma^2 = \frac{S(\hat{b})}{N-r-s} \tag{3.65}$$

For a particular estimate of the delay, d, we can calculate the variance, then choose the best estimate based on

$$\sigma^2_{min} = \min[\sigma^2_d \ , \ d = 0, 1, 2, \ldots] \tag{3.66}$$

It should be noted that the data will tend to limit the number of choices for the value of d to two or three possibilities.

Consider the following example. It was required to determine the values of the coefficients that describe the transfer function for the vertical (upper arm) axis. A step input of 100 D/A units was applied at time zero. The discrete step response data were then recorded, beginning at time zero. The sample rate was experimentally determined to be 9.6 ms, or about 104 Hz. This was also the rated sample rate of the transducer [Lord Corporation, 1987], which suggests neglible computation delay. This issue is

discussed further in Chapter 5. Using 200 samples, the X matrix was determined. Applying Equation 3.66

$$\begin{aligned} \sigma_3^2 &= 0.03177 \\ \sigma_4^2 &= 0.02954 \\ \sigma_5^2 &= 0.02997 \end{aligned} \tag{3.67}$$

which shows that a delay of four (d=4) minimizes the variance. The coefficients for r=3 and s=3 were then found to be

$$\begin{aligned} \alpha_1 &= 1.3135 \pm 0.00998 \\ \alpha_2 &= 0.0752 \pm 0.01817 \\ \alpha_3 &= -0.4066 \pm 0.00863 \\ \beta_1 &= 0.0200 \pm 0.000892 \\ \beta_2 &= 0.0037 \pm 0.001274 \\ \beta_3 &= 0.0242 \pm 0.001035 \end{aligned} \tag{3.68}$$

The decision concerning the choice of values for r and s was based solely on statistical considerations. The corresponding discrete open-loop transfer function for the vertical axis can thus be expressed in the z domain as

$$\frac{C(z)}{M(z)} = \frac{0.02\,z^2+0.0037\,z+0.0242}{z^4(z^3-1.3135\,z^2-0.0752\,z+0.4066)} \tag{3.69}$$

where z represents the familiar z transform operator, commonly used in digital control.

The step response predicted by this model, as well as the actual values determined by experiment, are plotted in Figure 3.8.

An important feature of this approach is the ability to quantify confidence in the model, as depicted by the confidence intervals shown above. The individual confidence limits give a fair estimate of the significance of the results. These confidence limits were computed using

$$b_i \pm t_\alpha(N-r-s)\sqrt{V_{ii}} \tag{3.70}$$

for i = 1, 2, 3, ..., k, where $t_\alpha(N-r-s)$ is the Student's t statistic for a $100(1-2\alpha)\%$ confidence region and $N-r-s$ degrees of freedom. V_{ii} is the diagonal element of the variance-covariance matrix V, found from

$$V = [X^T X]^{-1}\sigma^2 \tag{3.71}$$

Strictly, this estimate is only meaningful for matrices whose off-diagonal elements are small, which is often not the case for dynamic systems [Bollinger and Duffie, 1988]. However, the small confidence intervals suggest that this model is quite adequate to describe the empirical data. For this example $t_\alpha(194) = 1.65$ for a 90% confidence interval.

An obvious question is how small should these confidence intervals be to produce an adequate model? This problem is addressed in Chapter 4. However, an intuitive answer is that the model must be adequate for use in designing an effective compensator. It is important not to fall into the trap of generating more and more

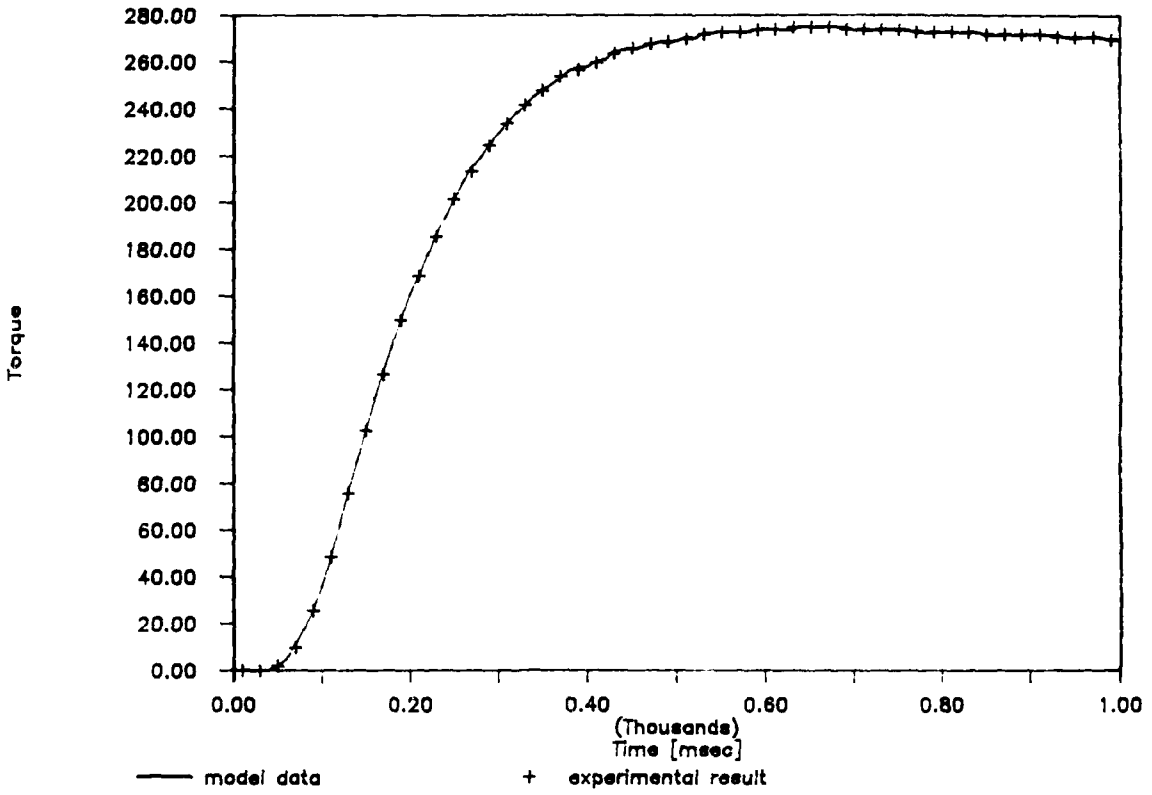

280.00
260.00
240.00
220.00
200.00
180.00
160.00
140.00
120.00
100.00
80.00
60.00
40.00
20.00
0.00
Torque
0.00
0.20
0.40
0.60
0.80
1.00
(Thousands)
Time [msec]
model data
+ experimental result

Figure 3.8

complex models, with the hope of improving their precision. For instance, an equation with two-hundred coefficients will fit two-hundred data points perfectly. However, the important question is whether it will facilitate a better compensator design than a model with one-hundred and ninty-nine coefficients, or one-hundred and ninty-eight? This seems unlikely. The objective, then, is to determine a model with just sufficient complexity to adequately quantify the transfer function, without generating physically meaningless, statistical precision.

3.4 Motion Control

Since a major goal of this research is to design a wrench control strategy that complements the existing motion (twist) controller, it is necessary to understand the existing robot motion controller. The GE P60 robot system consists of three main components as shown in Figure 3.9. The robot arm itself consists of joints, linkages, electric motors and transmissions which serve to locate and orient the robot end-effector. The teach pendant is used to teach program points and logic conditions for program execution. The RC 1560 controller consists of the electrical control, power amplifiers, and data processing and display modules for program control and execution.

The electrical control architecture itself is depicted in Figure 3.10. The motion control centers around the Central Processing Unit (CPU) and its associated electronics. The servomotors are controlled by the drive amplifiers on the basis of position feedback information received from the incremental encoders. These drive amplifiers then drive the motor according to the difference between a command velocity signal (voltage) issued by the CPU and the actual velocity signal received from the tacho-generator located on the motor. Also, relays on the servo amplifiers interlock with the controller's relay to preclude an overcurrent (current saturation) condition.

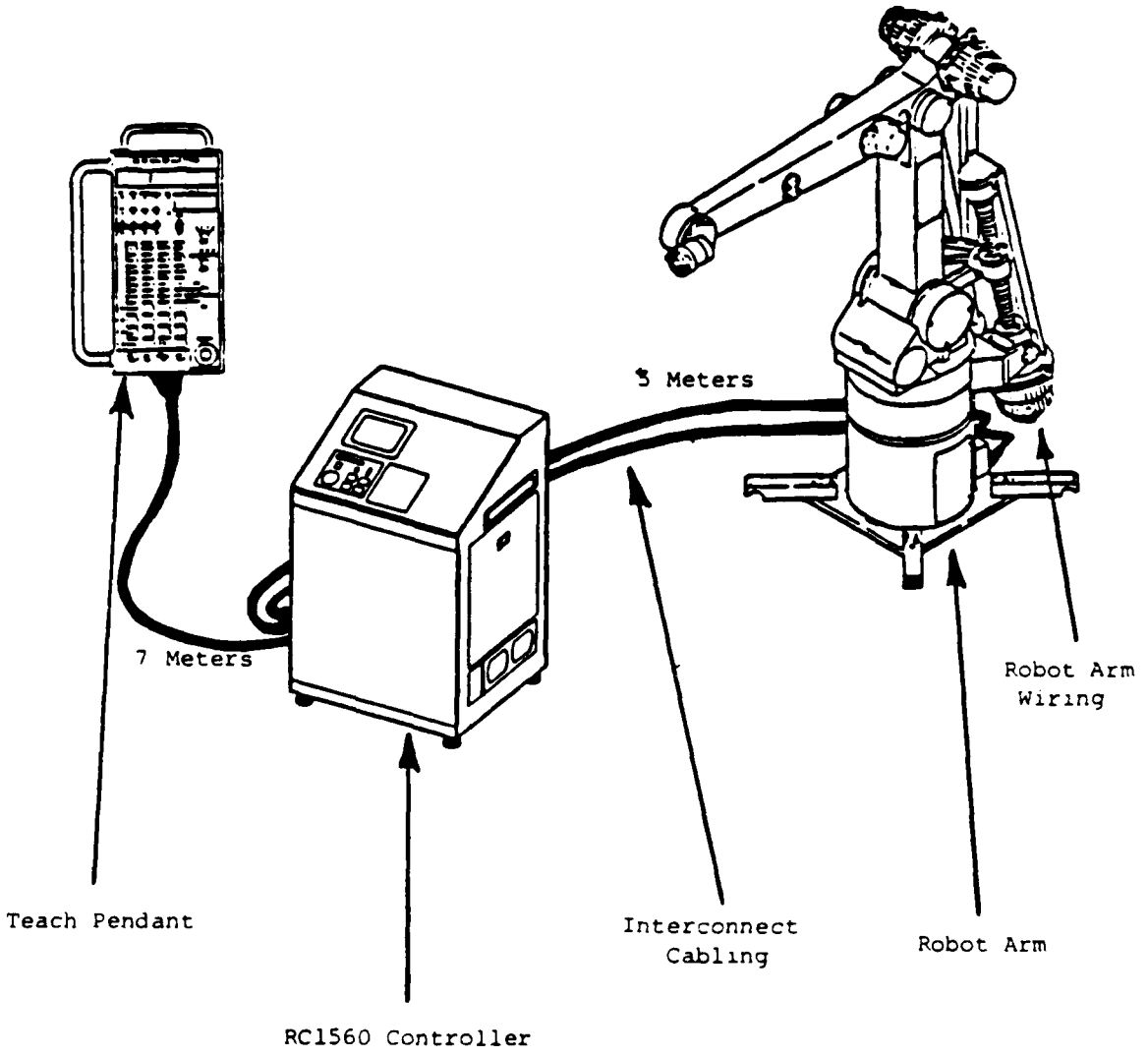
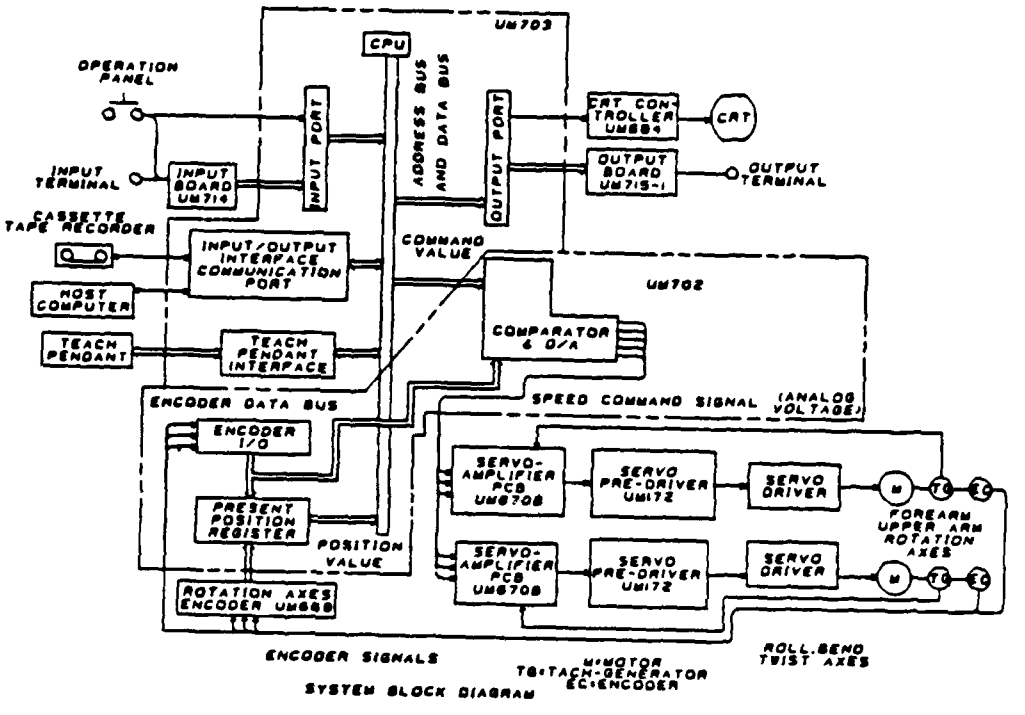

5 Meters
7 Meters
Robot Arm
Wiring
Teach Pendant
Interconnect
Cabling
Robot Arm
RC1560 Controller

Figure 3.9

SYSTEM BLOCK DIAGRAM



SYSTEM BLOCK DIAGRAM

Figure 3.10

A generic block diagram of the motion control circuitry is shown in Figure 3.11. The functions of these blocks are described as follows:

(1) The Path Processor contains algorithms which perform trajectory interpolation and coordinate transformations. Trajectory computations are performed to permit the axes to execute a coordinated motion.

(2) The Axis Processor performs the mid-level control functions by closing the motor position control loop. Like the path processor, this level is digital, interpolating as necessary between commands issued by the path processor. This block uses the difference between the commanded and actual motor positions as input. With this data, and the programmed velocity for a given step, the block generates a velocity command in the form of an analog voltage signal.

(3) The Servomotor block contains a velocity controller that minimizes the difference between the commanded and actual motor velocity. This correction signal is sent to the power amplifiers which drive the motors, themselves. The actual motor speed is measured by means of a tacho-generator which, like the incremental position encoder, is mounted directly on the servomotor. It is this block which contains the analog portion of the plant.

It is important to note that the transmission and load dynamics are outside the control loop. That is, the sensors (incremental encoders and tachometers) are colocated with their respective actuators (motors). This helps to ensure a very stable servo control loop, as discussed in Chapter 2. However, because there is no feedback from the end-effector, its actual trajectory can only be inferred from the forward kinematics.

BLOCK DIAGRAM OF MOTION CONTROL CIRCUITRY

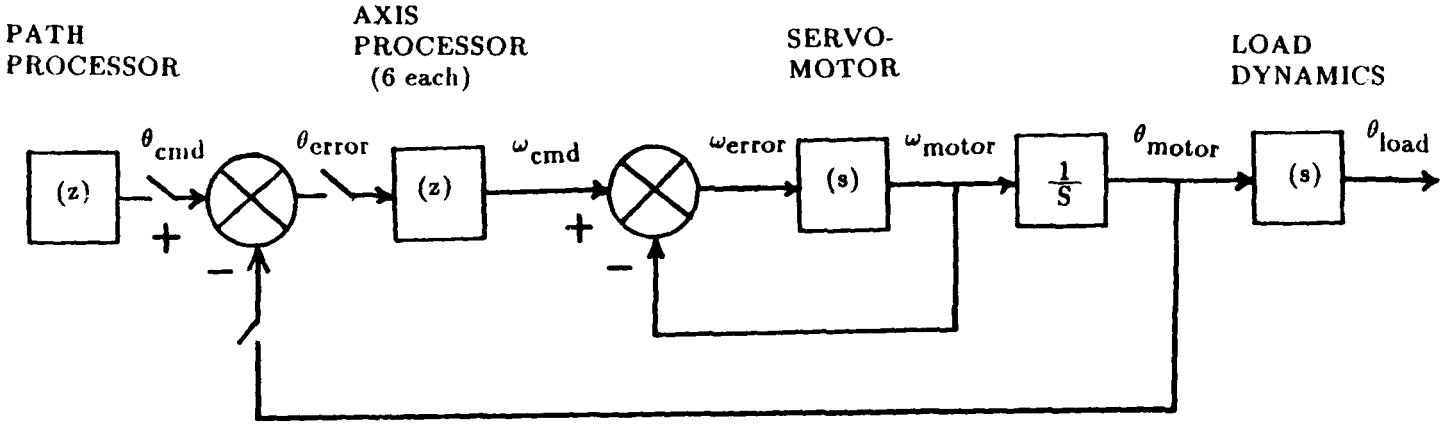


Figure 3.11

This lack of feedback is characteristic of industrial robots, and helps explain the need for heavy, rigid linkages (with their consequently low payload-to-weight ratios) to ensure end-effector positional repeatability.

Like all voltage-controlled robots, the GE P60's servomotors will draw whatever current is required to null the velocity error. If obstructed, this process continues until either motor failure, or a protection circuit shuts down the system. This protection circuitry in the GE P60, which is very reliable and quite sophisticated, was left intact, in order to maintain its safety features. Most force control strategies presume it is possible to draw an infinite current, which would necessitate the defeat of such circuitry; i.e. if they were ever to be used on a commercial system.

The cross-coordinated control strategy suggested here required no such modification. Furthermore, the individual feedback control algorithms were not only easy to implement, but facilitated the tuning of individual axis compensators for optimal performance, while avoiding the need to drastically alter the existing controller architecture or its circuitry.

3.5 Results and Conclusions

This chapter has described the kinematics, dynamics, and motion control system found on the GE P60 robot. Beginning with the kinematics, the reader was introduced to a notation [Duffy, 1980] which more readily facilitates the description of the manipulator kinematics in screw coordinates. The reverse analysis technique [Lipkin and Duffy, 1985b] was then applied to the GE P60. This approach to the inverse kinematics problem generates all solutions for a given position and orientation specification. Further, the forward solution was described.

The chapter continued with a discussion of the pertinent dynamics for industrial robots. A stochastic modelling approach for generating an axis transfer function was presented, including an example for the forearm axis.

Finally, the motion controller for the GE P60 was described. The main components were named and their functions cited. A generic block diagram for this type of industrial robot was also presented and discussed.

CHAPTER 4
CONCEPTUAL DESIGN

4.1 Introduction and Objective

The purpose of this chapter is to present a conceptual design methodology. Sufficient detail is provided such that this methodology could be duplicated with little difficulty. Other issues relating to hardware and systems development are presented in Chapter 5.

An important characteristic of a good design process is to work from the top down, so as not to be committed too early to any of the lower level details. The organization of the control signal flow is described, beginning with a discussion of the control architecture. As previously discussed, this architecture, which is based on external control loops, was chosen because it permitted the augmentation of existing circuitry, without the need to defeat the built in safety features. Additionally, related issues such as disturbance rejection and the choice of coordinate systems are discussed.

The next section describes the central issue of the controller design process, namely the design of the compensator. A step-by-step procedure, including a detailed example from this research, is described. Performance results for this compensator are also presented.

The most esoteric topic of this chapter is that of constraint formulation. This process can be treated separately from the servo control problem, primarily due to the choice of the control system architecture. Background for this process was presented in

Chapter 2 in the sections on screw theory and kinestatics. Examples of how this process can be applied to formulate the constraint wrenches for several common industrial tasks are presented. These example tasks were implemented in the laboratory, as described in Chapter 6.

4.2 Control Architecture

The feedback of end-effector wrench data to the robot controller provides an effective means of compensating for robot repeatability, tool wear and variations in workpiece geometry or fixturing. This ability to control the relative position of the end-effector through wrench sensing makes these task positioning uncertainties much less significant. While this capability is extremely powerful, most robot controllers provide no facility to integrate task space sensor data, as shown in Figure 4.1a. Here the motion controller computes the desired axis set points in order to satisfy the position and velocity program data through the use of coordinate transforms and interpolation algorithms. Individual joint commands are then sent to the axis controllers, one for each axis. These controllers close position and velocity control loops around individual axes. No end-effector feedback is included. Without such feedback, however, many close tolerance tasks are not practical, while others require very careful use of teach pendants for path programming and touch-up programming, virtually eliminating any possibility of off-line programming or integration with CAD/CAM systems. Two alternative architectures for external control loops have been put forth in the literature [Sweet and Good, 1984], as shown in Figure 4.1b and Figure 4.1c. The addition of task space feedback, however, can be difficult to implement.

One major design challenge results from the relatively slow sample rates associated with force sensors (on the order of 100 Hz). By incorporating force feedback

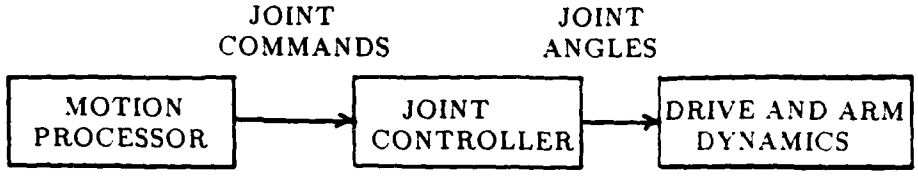


Figure 4.1a

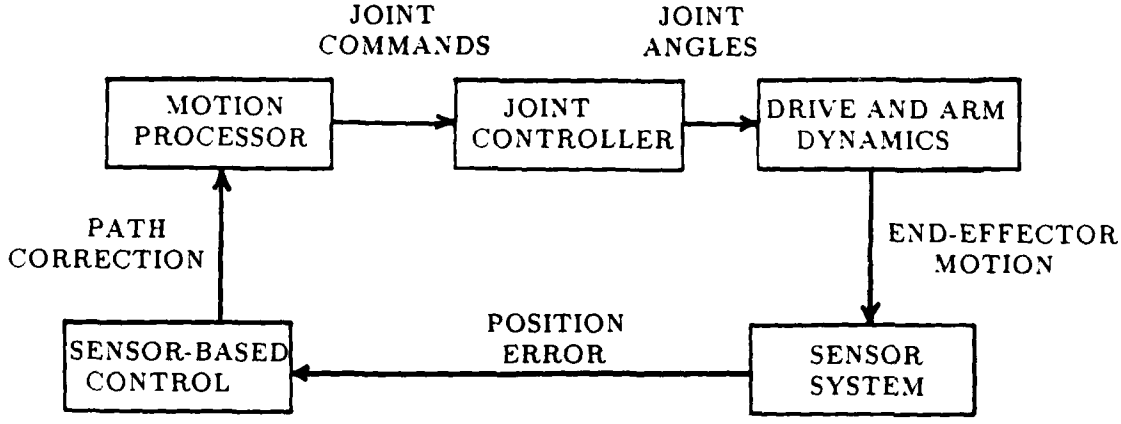


Figure 4.1b

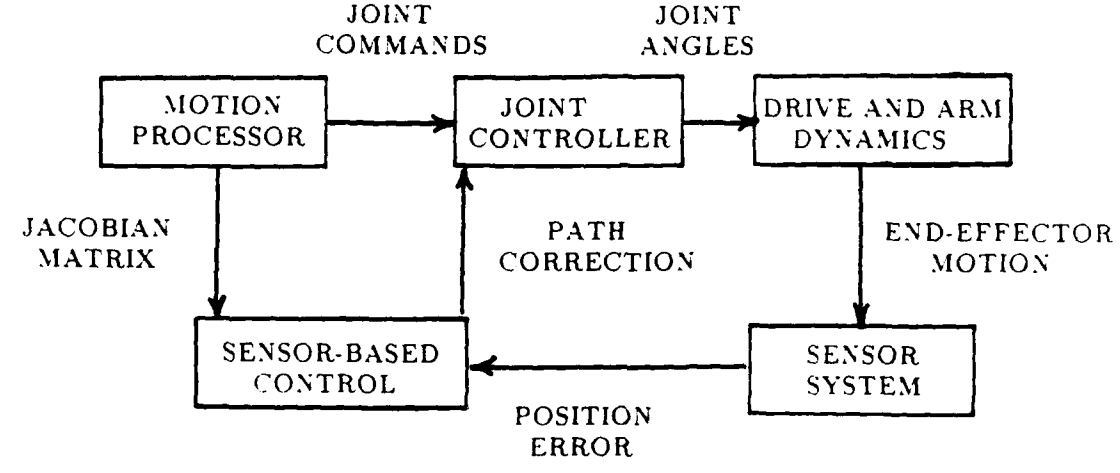


Figure 4.1c

Figure 4.1 Robot controller configurations.
a)Open-loop: b)Feedback at the motion processor: c)Feedback at the joints.

into a single rate controller, as is often suggested, the servo rate will be limited by the data transfer rate. Recall that the permissible loop gain is a strong function of servo rate. Thus it is desirable that this rate be as high as possible to ensure satisfactory performance and disturbance rejection characteristics. One should note that position servos commonly used on commercial robots have servo rates of from 500 to 2000 Hz. By utilizing a split rate controller architecture, with a 100 Hz motion controller and 1000 Hz axis controller, the potential gain limitations of the GE P60, associated with the slower servo rate of the motion controller, are avoided. The architectures proposed in Figures 4.1b and 4.1c similarly minimize the effects of servo rate limitations associated with the use of a force sensor, by also using split rate control.

As previously mentioned, both performance and disturbance rejection, including the effects of friction and backlash, are strongly affected by servo rates. While these disturbance effects have been greatly reduced by improved mechanical design, they can never be completely eliminated. The most effective way to deal with them is to enclose these disturbances in a high gain loop, namely the position loop. This loop characteristically has a much higher gain than the force loop due to the colocated sensor, as discussed in the section on dynamic stability in Chapter 3, and a faster sample rate, as stated above. Using a single rate servo loop closed with a non-colocated force sensor dramatically limits the maximum stable loop gain. Nonetheless, some force control strategies have tried to implement a single rate control architecture which includes the use of a force sensor [Salisbury, 1980, and Raibert and Craig, 1981].

For the method depicted by Figure 4.1b, path correction data is fed to the motion controller, which must then compute the corrected robot trajectory and subsequent joint commands. For this structure, the essential control signal flow remains

unchanged. The motion processor block remains in the forward loop, as depicted in Figure 4.2.

While the ability to leave the control circuitry essentially intact is appealling, it is often nearly impossible for anyone but the manufacturer to actually implement this architecture. Attempts by others usually fail, since the details of the control circuitry, particularly the bus architecture and data transfer protocol, are invariably proprietary. As such, vendors will rarely supply sufficient information for a user to interface directly with the motion controller, precluding the use of this approach. This leaves the architecture of Figure 4.1c as the only practical alternative for most users, and hence the choice used here.

For this structure, the sensor-based controller provides signals that directly modify the servomotor commands. As such, the motion controller is effectively moved from the forward path to the feedback path, as shown in Figure 4.3. A side benefit is that if the force controller servo rate is faster than the commercial motion controller, there exists the potential for a higher control bandwidth. More significant, however, is the relative ease of implementation.

For voltage-controlled robots, like the GE P60, the velocity command is physically manifested as an analog voltage as shown in Figure 3.12. Because of the modular design of modern electronics, these voltage inputs to the servo amplifiers are usually fairly accessible. By adding a voltage signal proportional to the reciprocal wrench constraint error, the motor velocity remains essentially unchanged, as does, therefore, the end-effector twist. The additional motor torque is used to overcome the increased static load due to the constraint wrench, as discussed in Chapter 2. The details of the additional circuitry required are included in Chapter 5.

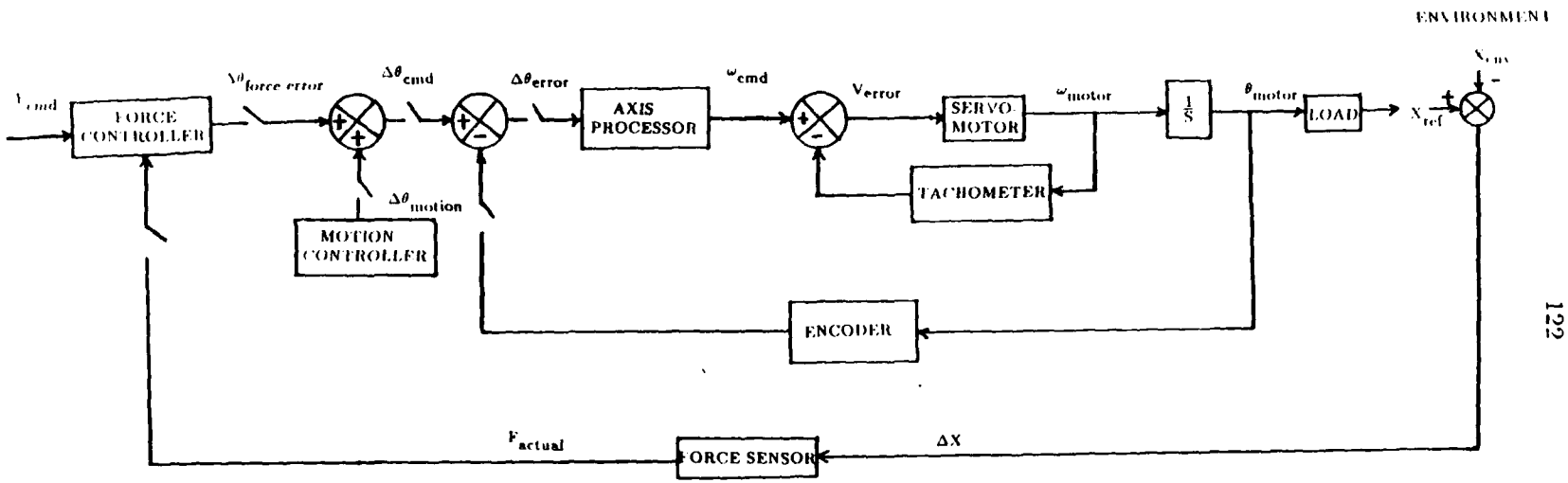

ENVIRONMENT
$\Delta\theta_{force\ error}$
$\Delta\theta_{cmd}$
$\Delta\theta_{error}$
ω_{cmd}
V_{error}
ω_{motor}
θ_{motor}
$\Delta\theta_{motion}$
F_{actual}
ΔX
FORCE CONTROLLER
AXIS PROCESSOR
SERVO MOTOR
LOAD
TACHOMETER
MOTION CONTROLLER
ENCODER
FORCE SENSOR

Figure 4.2

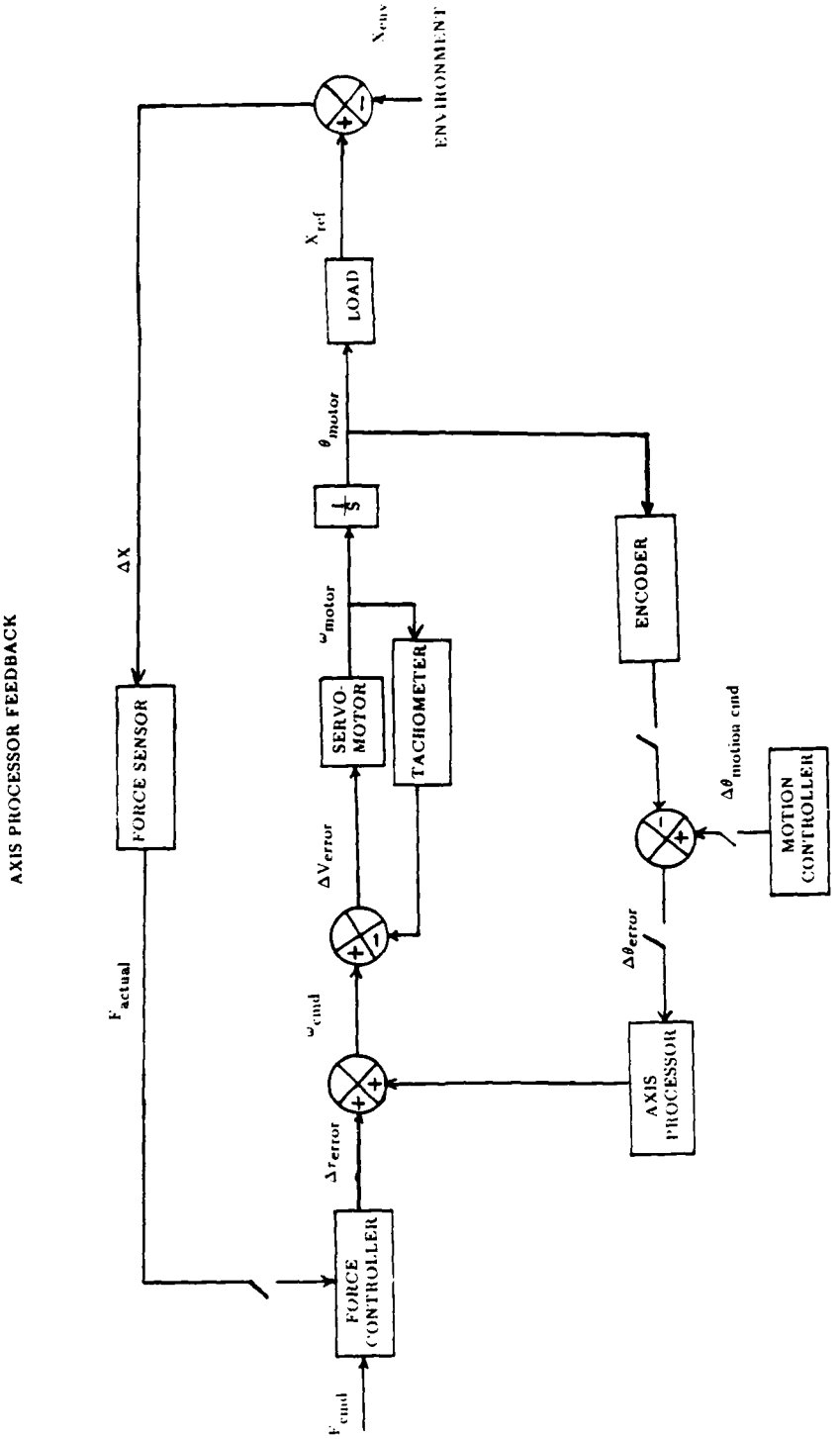

AXIS PROCESSOR FEEDBACK
ENVIRONMENT
LOAD
ENCODER
SERVO-MOTOR
TACHOMETER
FORCE SENSOR
FORCE CONTROLLER
AXIS PROCESSOR
MOTION CONTROLLER
ΔX
θ_{motor}
ω_{motor}
ΔV_{error}
F_{actual}
ω_{cmd}
$\Delta\tau_{error}$
F_{cmd}
$\Delta\theta_{error}$
$\Delta\theta_{motion\ cmd}$

Figure 4.3

It is worth noting that this architecture lends itself well to Cartesian based control, which is desirable. Cartesian control strategies are those where the error is formed in task space, as opposed to joint space. Such strategies have the advantage that the constraints, as discussed later in the chapter, are more naturally formulated in Cartesian space [Mason, 1981, and Lipkin and Duffy, 1987]. In joint based controllers, the Cartesian errors are never directly formed, thus clouding the distinction between constraint formulation and the lower level, servo control issues.

The main attraction of a joint based controller is its apparent simplicity, due to the lack of any need to perform coordinate transformations. This suggests the potential for greater speed. However, experience has generally shown that by the time sufficient functionality has been added to joint based controllers, such as Salisbury's stiffness controller [Salisbury, 1980], the difference in the level of computational burden disappears [Maples and Becker, 1986]. What ever time penalty remains is offset by the simplification of constraint formulation.

4.3 Compensator Design

The most important issue in compensator design is the availability of an accurate model of the plant to be controlled. It is clear from Figure 4.3 that this plant has both discrete and continuous elements. However, in order to apply standard root locus design techniques in the discrete domain, it is necessary to reduce these elements to a single, discrete transfer function which accurately characterizes this plant. This procedure was completed in Chapter 3. Figure 4.4a is a z-plane root locus of the vertical (forearm) axis based on the model presented in Chapter 3. Figure 4.4b shows that the system becomes unstable for a loop gain of about 1.5, due to the lightly damped poles. Figure 4.5 shows the simulated, closed-loop, unit step response for a loop gain of 1.5.

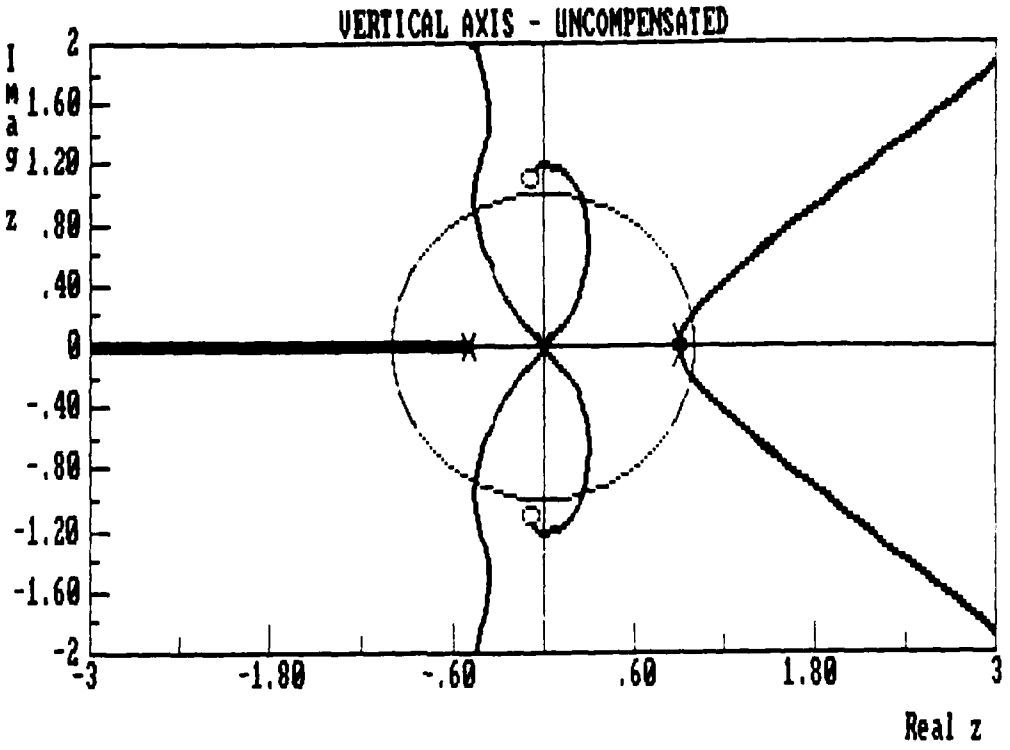


Figure 4.4a

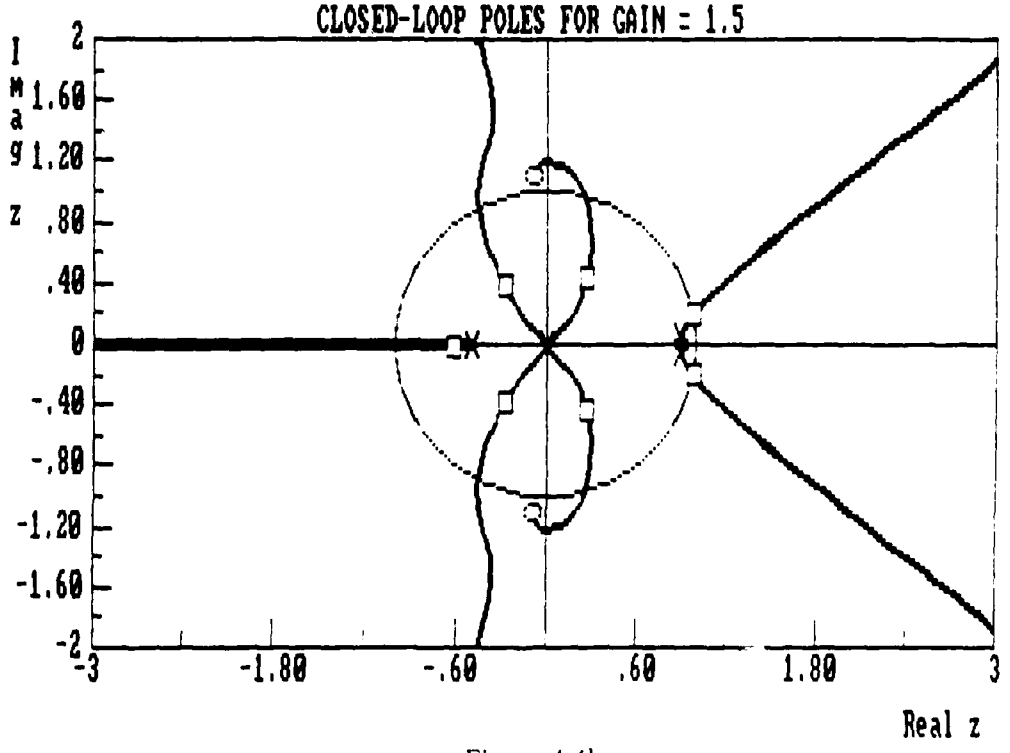


Figure 4.4b

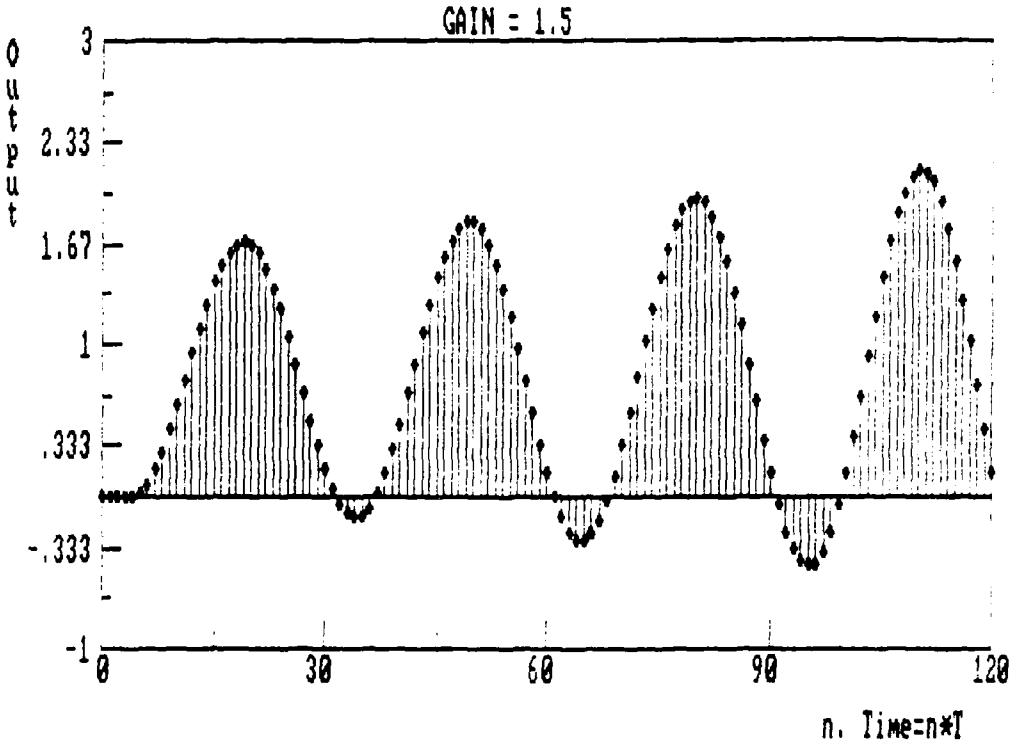


Figure 4.5

The purpose of the compensator is to move the closed loop poles to a region of greater stability and better performance. Sweet and Good [1984] suggested the following criteria for step response performance

- The rise time should be as short as possible
- The initial overshoot and undershoot should not exceed 10% of the final value, and ideally be zero.
- The settling time should be as short as possible.

Most performance results reported in the literature provides step response data [Maples and Becker, 1986, Haefner et al., 1986, and An et al., 1988]. They typically report overshoots of 25% to 100%. Such high overshoot means that the servo loop has a very low damping ratio. This characteristic calls into question the ability of such systems to perform contact tasks in a rigid environment, as is characteristic of an industrial setting. This low damping is particularly unsuitable for close tolerance, assembly tasks with bilateral, geometric constraints, such as for the peg-in-the-hole type of problem.

Therefore, in choosing between response time and overshoot, overshoot must be the limiting design consideration. One interesting experimental result observed was that when the compensator design was tested, the system was more highly damped than simulation suggested. This was apparently due to frictional damping. Because of this, the author found that designing a controller with 5-10% overshoot in simulation resulted in an actual system with virtually no overshoot.

Different forms of compensators were considered, including state space designs, adaptive control designs and conventional proportional plus integral plus derivative (PID) designs. State space formulations have the great advantage of

being able to conveniently handle multiple input/multiple output (MIMO) systems. However, in servo applications, their implementation is computationally burdensome as shown in Ogata [1987]. Given that the robot actuators have been shown to be effectively decoupled (essentially a collection of single input/single output (SISO) servo loops), the complexity and resulting computational burden of a state space formulation tended to outweigh the potential advantages of theoretically being able to arbitrarily place the closed-loop poles. Furthermore, physical system limitations, such as servo bandwidth and torque limits, also precluded full realization of this theoretical capability.

Adaptive control was also considered, but similiarly rejected. Craig [1988] reported experimental results for an adaptive control strategy he implemented on a *two link industrial robot, the Adept One.* Even without force control, the computational burden was such that Craig recommended the use of a dedicated microprocessor for each joint axis, each with the approximate power of a Motorola 68000, in order to successfully implement his algorithm. No doubt force control would further increase this computational burden. Hence, the use of an adaptive control strategy was rejected for this research.

Proportional plus integral plus derivative (PID) controllers have been used effectively for years as analog process controllers, and their use is familiar to a broad cross section of mechanical, electrical, industrial and agricultural engineers. The proportional term governs response time and robustness to disturbances. The derivative term induces a stabilizing effect by contributing to the damping, while adding an anticipatory effect to the error signal. The integral term, while destabilizing, is needed to eliminate steady state error to a constant command value.

The dc servo motor is usually considered to be a type one system (possessing a free integrator), because the system input voltage is proportional to the derivative of the output variable, position. This is not the case when an external control loop is introduced. The addition of the external control loop reduces the plant to type zero. That is, each axis becomes a type zero system, requiring the integral effect of the PID controller to eliminate steady state error.

For those generally familiar with analog PID control, but rusty or unfamiliar with its use for discrete systems, Bollinger and Duffie [1988] offer an extremely straight forward development for discrete PID control. Certainly other models are available. Åström and Wittenmark [1984], Ogata [1970] and Franklin and Powell [1980] list several each. Maday [1987] also presents a novel connection between classical and modern control designs. However, the straight forward development of the discrete PID controller by Bollinger and Duffie [1988] is quite adequate for this application, as well as being very simple to implement.

Using a digital computer for control, one finds it very convenient to express the correction signal in terms of a change in value, rather than an absolute value. Using this convention

$$\Delta m_n = m_n - m_{n-1} = K_0 e_n + K_1 e_{n-1} + K_2 e_{n-2} \tag{4.1}$$

where Δm_n is the change in system input (correction) at time n, which is computed based on the error, e, the proportional gain, K_p, the integral gain, K_i and the derivative gain, K_d, and the sample interval, T, where

$$K_0 = K_p + K_i T + \frac{K_d}{T} \tag{4.2}$$

$$K_1 = -K_p - \frac{2K_d}{T} \tag{4.3}$$

$$K_2 = \frac{K_d}{T} \tag{4.4}$$

This type of controller allows one to arbitrarily place two zeros. The two poles, however, are placed at z=0 and z=1 in order to include the stable portion of the real axis in the locus. The free integrator (z=1) ensures a zero steady state force error for a constant force command as discussed earlier. The zeros need to be located so as to draw the lightly damped poles inward to a more desireable position.

Unlike Stepien et al. [1987], who used a compensator with two zeros at 0.82

$$C(z) = \frac{(z - 0.82)^2}{z(z - 1)} \tag{4.5}$$

for the smaller wrist axes, and a compensator with two zeros at 0.7,

$$C(z) = \frac{(z - 0.7)^2}{z(z - 1)} \tag{4.6}$$

for the larger axes of the GE P50, swing, horizontal and vertical, the author found that a single compensator based on

$$C(z) = \frac{G(z - 0.9)^2}{z(z - 1)} \tag{4.7}$$

where G is a gain to be determined for each axis, provided both ease of performance tuning and excellent response. For the vertical axis with G=1, the step response is somewhat sluggish, as shown in Figure 4.6 Increasing the loop gain, G, to 3.3 dramatically improved performance, as shown in Figure 4.7. Continuing to increase the loop gain reduces the rise time and improves disturbance rejection even further, but at the expense of increased overshoot and settling time. Finally, a loop gain of 10.0 caused the system to become unstable, as shown in Figure 4.8. The closed-loop unit step response for the uncompensated system is shown in Figure 4.9. This clearly indicates that the compensator facilitated the use of a higher loop gain than could otherwise have been used. The corresponding root locus of this compensated system is shown in Figure 4.10. Simulation proved to be a useful development tool to determine the proper range of compensator gain.

One should also note that this type of simulation, which uses a discrete model of the system similar to the one developed in Chapter 3, and a discrete compensator, can easily be duplicated on a modest computer system. While many excellent and quite sophisticated simulation software packages are available, they are not needed to duplicate the procedure described here.

Having developed the model of the plant, and a compensator to control it, one can then address the problem of generating appropriate command values of the constraint wrench which will facilitate a successful completion of the task. This process is referred to as constraint formulation.

4.4 Constraint Formulation

It is important to recognize that the constraint formulation process is strictly separate from the servo control problem. This separation greatly simplifies the

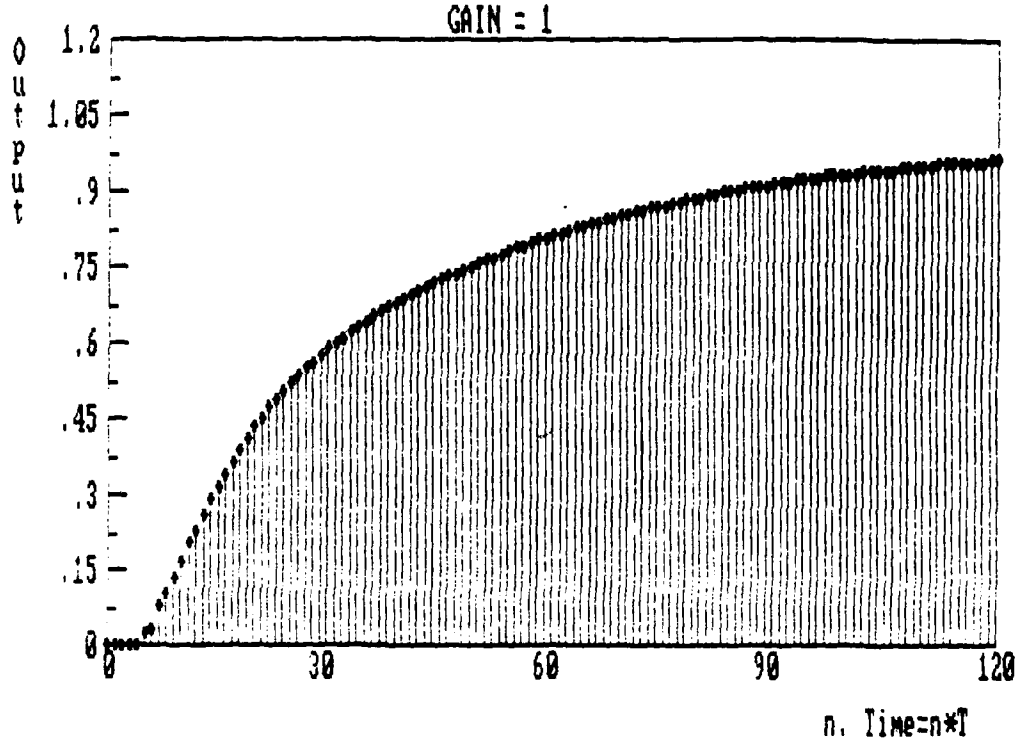


Figure 4.6

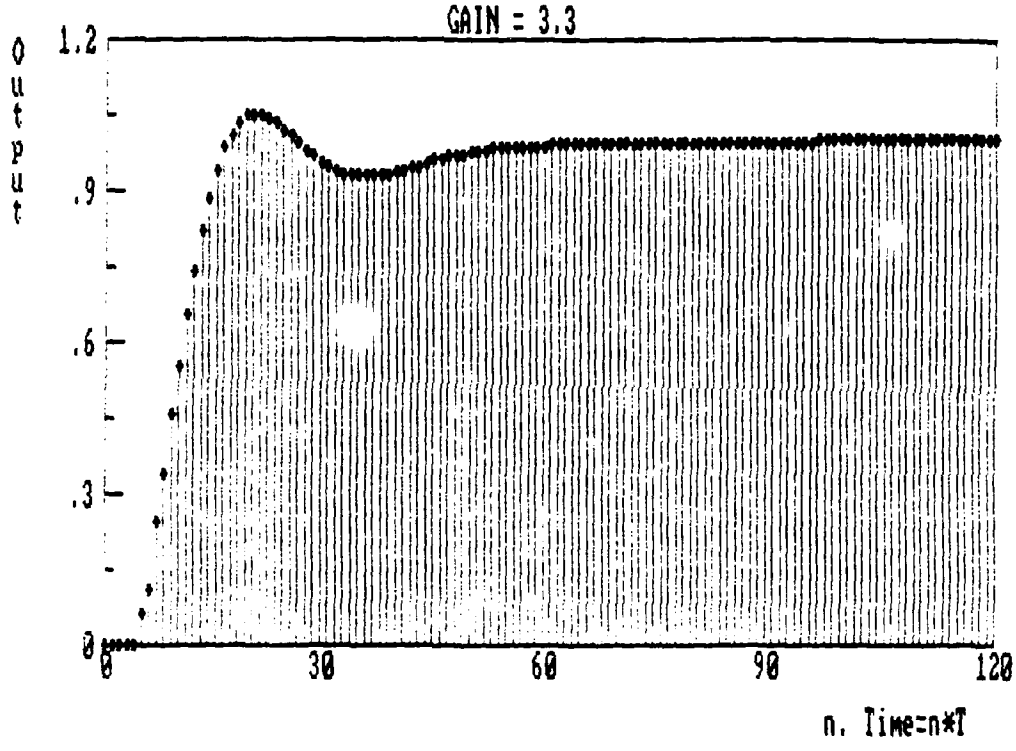


Figure 4.7

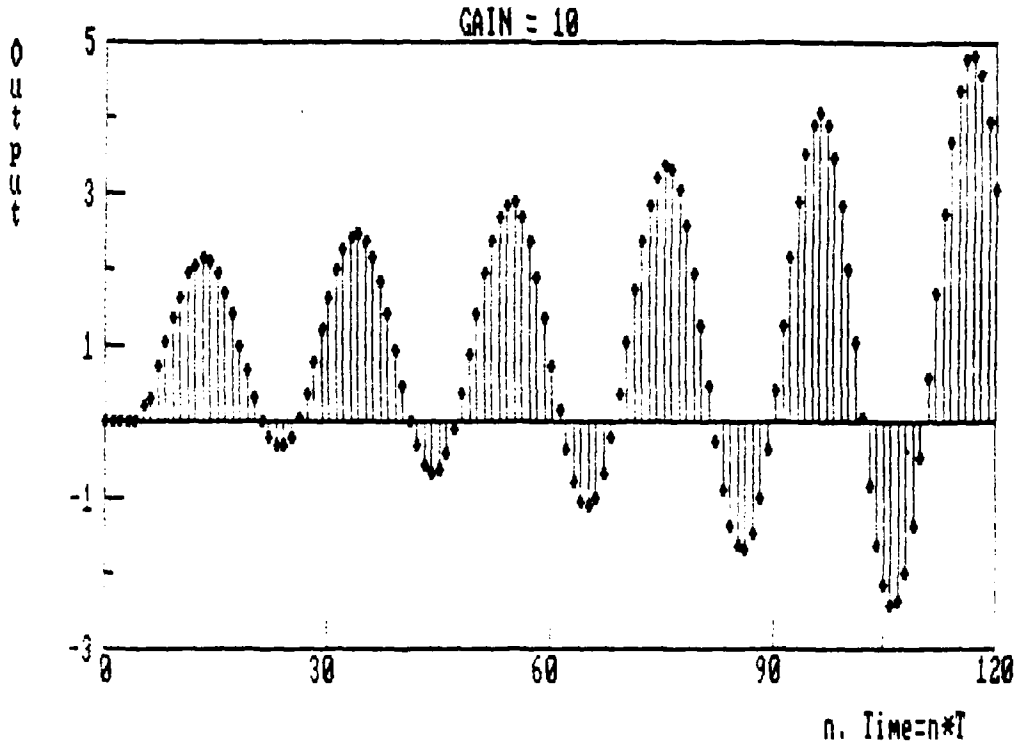


Figure 4.8

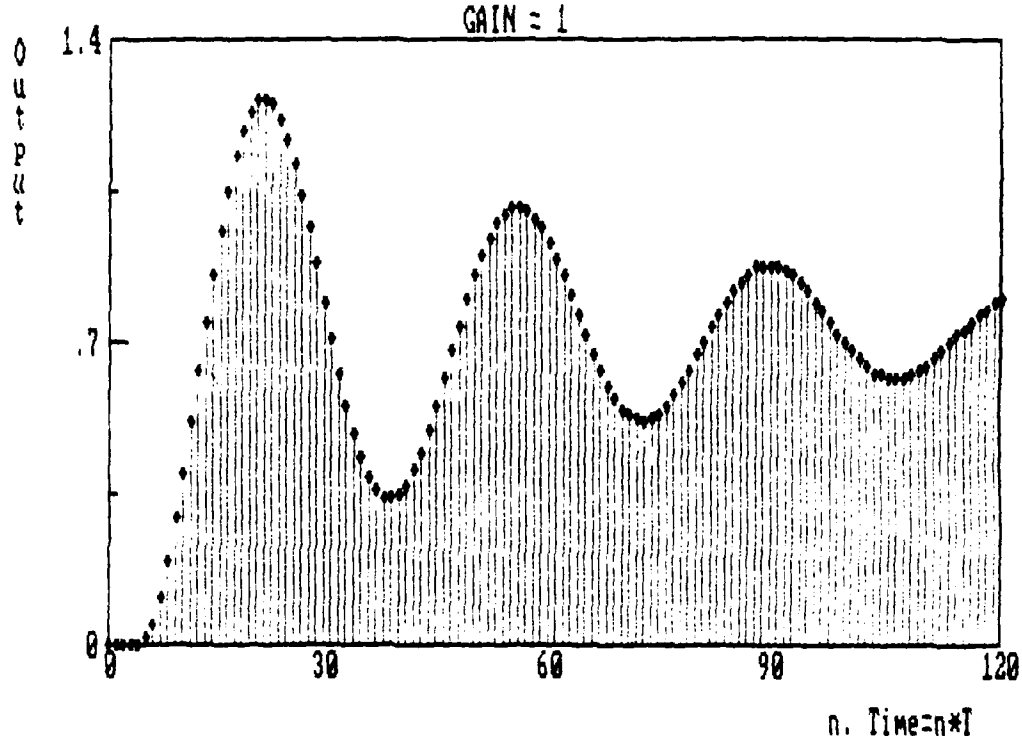


Figure 4.9

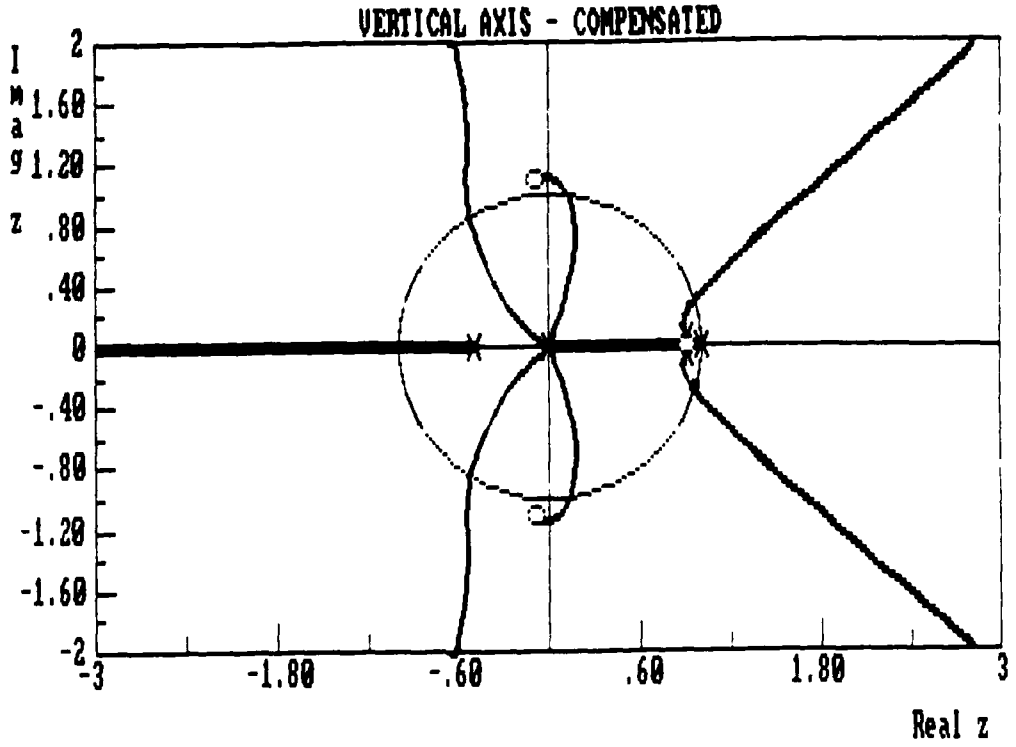


Figure 4.10a

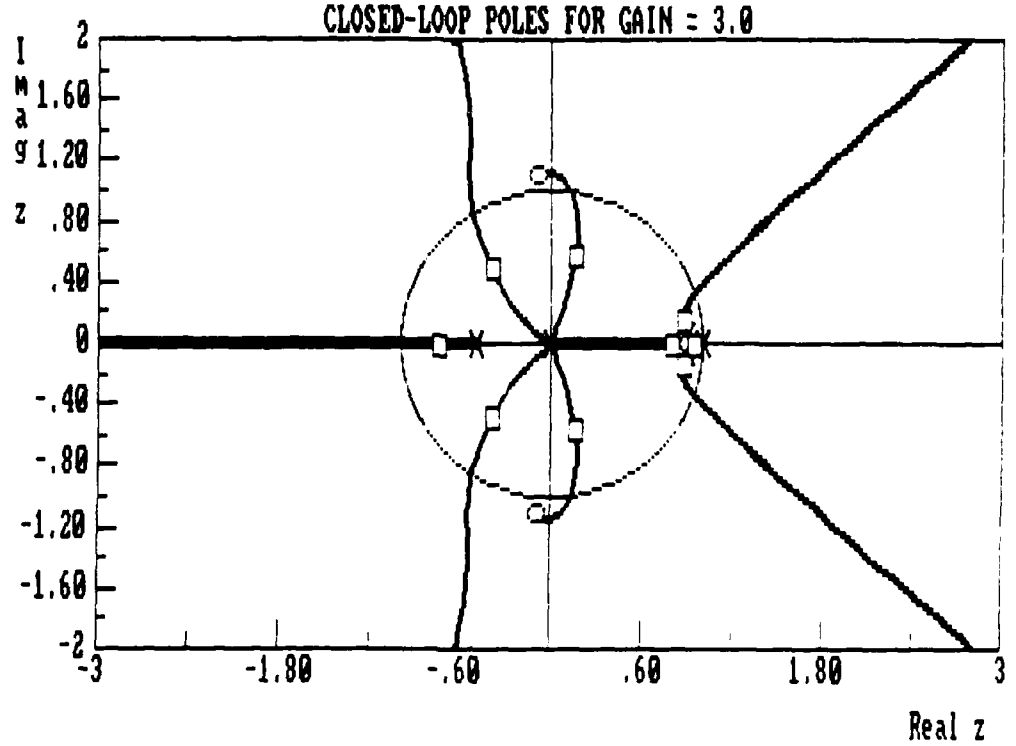


Figure 4.10b

implementation of new tasks for a given robot system. Since the compensator design is independent of the command, only the constraint formulation problem need be addressed in order to perform a new task. This type of functional separation is indicative of the approaches that derive from Mason [1981], such as those reported by Paul and Shimano [1976], Raibert and Craig [1981] and De Schutter [1986].

Most earlier formulations used the feedback matrix, itself, as a means of constraint specification. These approaches included damping control [Whitney, 1977], stiffness control [Salisbury, 1980] and impedence control [Hogan, 1985, and Kazerooni et al., 1986] as described in Chapter 1. Unfortunately, tuning the gains to achieve some apparent stiffness or impedence characteristics means that the servo performance will invariably be less than optimal, and usually much less [Whitney, 1987]. Even so, the principal difficulty with using such formulations is that the feedback matrix is only good for one set of constraints. Thus, a new controller has to be designed explicitly for each new constraint specification. Consider also that complex assembly jobs may include several tasks, each with a different set of constraints. Mason [1983] termed the use of the controller in this fashion "explicit feedback", as compared to the more powerful, decoupled constraint specification techniques which sprang largely from his work. These other methods have been termed the "hybrid control functional specification mechanism" [Mason, 1983]. It should also be noted that Mason uses the term "hybrid control" to refer to any approach that formulates the constraints in a similiar fashion, without regard to the specific details of a particular implementation.

It is important to recognize that the procedure presented here is different from all the other approaches of this type. The constraint formulation is based on a

sound, geometric foundation. Raibert and Craig [1981] used a diagonal selection matrix, S, based on Mason's work [1981]. This formulation has already been shown to be non-invariant with respect to a change of origin, basis or scale, not to mention kinematically unstable for most anthropomorphic, industrial manipulators. De Schutter [1986] recognized that Mason's theory was limited to certain ideal task geometries, unsuitable for most real world applications. Friction, finite stiffness and geometric uncertainties, among other things, made Mason's idealized formulation largely impractical .

Mason [1981] assumed that every task could be divided into either position (velocity) controlled or force controlled directions, where these directions are orthogonal and complementary. This type of specification, which is based on an erroneous notion of orthogonality, is not origin invariant, and therefore very sensitive to the choice of coordinate system used to describe the task. As such, De Schutter and Van Brussel [1988] formalized a coordinate system selection process by proposing five alternatives:

1) global reference frame (grounded)
2) object frame (alligned with the "natural constraints")
3) task frame (a computer representation of the physical object frame)
4) robot end-effector frame
5) robot base frame

where a frame is defined as a relative position and orientation with respect to some well defined reference, typically using homogeneous transformation matrices to store the data describing this relationship.

They, like Mason, correctly recognized that different tasks are more easily specified in one frame than another. The major weakness of this approach, however, is the use of an inner product that is origin dependent. Specifically, De Schutter and Van Brussel suggest the following approach:

> Determine position- and force-controlled directions assuming ideal circumstances (infinite stiffness, no friction) and/or simplified task kinematics, and design a control implementation which is robust with respect to the discrepancies between the real and ideal world (i.e., robust with respect to the occurrence of motion in force-controlled directions and of forces in position-controlled directions). Of course, the more a task description approaches the real task environment, the more the performance approaches the desired one. [De Schutter and Van Brussel, 1988, page 5]

This relatively simple, no nonsense approach is what one would expect of work which has been experimentally tested, as De Schutter [1986] did on an electro-hydraulic industrial robot, the Cincinati Milicron T3-556. However, the simplicity and utility of such an approach can be dramatically enhanced by the use of an origin invariant inner product for constraint specification, namely the reciprocal screw product.

De Schutter [1986] still relied on what is essentially a selection matrix in order to specify "directions" to be either velocity controlled, force controlled or "tracked". For those physically unconstrained degrees of freedom, he suggested the specification of a diagonal, end-effector freedom matrix, M_F, which is essentially the same as the selection matrix of Raibert and Craig [1981]. This seems to be used primarily for problems like contour tracking, where the goal is to maintain tracking in a "force direction", but unconstrained directions are present in the task frame. For tasks which are *primarily* limited by the task geometry, such as assembly tasks, he suggests a diagonal, task frame constraint matrix, M_C, to select motion constraints.

Clearly, this type of constraint formulation suffers from exactly the same limitations as those of the other orthogonal approaches, despite its experimental demonstration. One must recognize that the degree of success relies heavily upon the "correct" choice of origin. That origin being one where the six degrees of freedom are essentially decoupled, the so-called "center of compliance." Such an ad hoc formulation lacks the elegance of a strategy based on sound mathematical principles. The formulation presented here retains the advantages of an independently designed, robust controller, but at the same time is free of the limitation of origin dependence in constraint specification, as was the the approach used by De Schutter and Van Brussel [1988].

Cross-coordinated control similiarly takes advantage of the robustness of the servo controllers, as do all the strategies which make use of external force control loops. However, instead of some geometrically meaningless approximation based on orthogonality, this strategy makes use of the geometric insight provided by screw theory to formulate geometrically meaningful constraints, based on reciprocal screws. This strategy satisfied the following requirements:

1) The constraint formulation (programming) problem is strictly separated from the control (compensator design) problem. In this way, the servo loops can be tuned for optimum performance, disturbance rejection and general robustness, independent of constraint specifications.

2) The constraint model should be as simple as possible, while still representing the actual task as accurately as possible. If friction is relatively small, such as when a roller bearing moving over a rigid surface, it should be ignored as far as the constraint formulation is concerned. If speeds are slow, as they almost certainly are for contact

tasks, then inertia, Coriolis and centrifugal (kinetic) terms may also be neglected. The determining factor is whether these impulsive wrench (or compliant twist) effects contribute to the sensor signal in a significant way. That is, does ignoring them significantly degrade task performance? If so, then the feedback signals will need to be decomposed and filtered, with an associated increase in the complexity of the implementation. The criteria is clearly subjective, but a threshold contribution of 10% error seems to be a good rule of thumb, and was used here.

3) Most important, the specified constraint wrenches must always be reciprocal to the instantaneous twist (velocity) commands. The reason for doing this is simple. Such a constraint specification is origin independent! Hence, the constraints are geometrically meaningful, and independent of what ever choice of origin is most convenient to describe a particular problem. This minimizes the need to rely on the underlying robustness of the servo controller for successful task completion.

Cross-coordinated control was intended to serve as an augmentation to the commercial control system for industrial robots, as earlier described. As such it should blend in unobtrusively, making use of built in capabilities whenever possible. A typical implementation would proceed as follows:

1) The operator teaches a series of points (program steps), using as few as possible, that follow the required nominal trajectory. Points should be connected by straight lines or arcs of circles, which is how such controllers are normally designed.

2) The operator specifies controller outputs to designate those program steps where wrench control is required.

3) The need for wrench or twist decomposition [Griffis, 1988] is evaluated and is included as required. Feedback signal decomposition is then implemented in software as indicated.

4) The programmer specifies a reciprocal constraint wrench for each program step where it is required, such that it will accomplish the task for that program step. This, too, may require some experimentation, particularly for some process tasks like grinding or deburring, where important design parameters are often empirically determined. This issue is discussed in some detail in Chapter 6.

5) The programmer plans for the possibility of execution failure, since any control system will be subject to faults which must be sensed and acted upon. The rule of thumb is to "fail safe". When a fault occurs, the system must revert to a condition that is not dangerous to personnel or equipment.

The question as to why the program steps (the path from one taught point to the next) were straight lines or circular arcs is now answered. Clearly, they need not be, but this choice certainly simplifies the constraint formulation problem. The line coordinates of the instantaneous screw axis which describes a straight line motion are ($\underline{0}$;$\underline{S}$). This axis lies along a line at infinity, throughout the step. An arc of a circle can be described as a screw whose instantaneous screw axis has coordinates ($\underline{S}$;$\underline{0}$). This instantaneous screw axis is at the center of the circle throughout the step. This means that for both cases, the finite screw and the instantaneous screw describing the twist are coincident. Also, they are lines, since they satisfy the quadratic identity. This makes the specification of reciprocal screws (wrenches) quite simple and straight forward. One should note that this generally involves no loss of capability, since modern commercial

robots are normally programmed to follow a straight line (or, less commonly, a circular arc) between taught points. Even interpolated motion, where tool orientation or tip position remain constant, exhibits this type of motion. Using these simple program steps, the operator can program any task that the robot is physically capable of.

Consider the following examples, which are indicative of the types of tasks required of industrial robots on the manufacturing floor. We begin with the peg-in-the-hole problem, as shown in Figure 4.11.

This problem is presented in virtually every paper on hybrid control. Once the tip of the peg is within the hole, the fully bilateral constraints make it a simple problem to describe. The fact that the initial portion of the problem, namely the tip insertion, is usually neglected is a by-product of the criticism which has been raised over and over, namely the lack of experimental work. Anyone who has performed this task in the lab knows that tip insertion is a significant part of the problem!

The author suggests here two alternative approaches for tip insertion, which were both implemented experimentally. The first is the use of a chamfered hole in conjunction with the RCC device, which is described in Chapter 5. As long as the chamfer width exceeds the total positional uncertainty of the task, which is the sum of robot repeatability, part geometry uncertainty and part location uncertainty, the RCC device ensures that the tip insertion is successful. This process is shown in Figure 4.12. No wrench constraint is required for this step.

When the chamfer is inadequate, or absent altogether, a constraint wrench is required. The author used a tipping strategy, as suggested by Seltzer [1986] and others. Figure 4.13 depicts the several program steps required. The peg, which is tilted at 10° to the vertical, approaches the hole. On contact, the robot initiates the next program

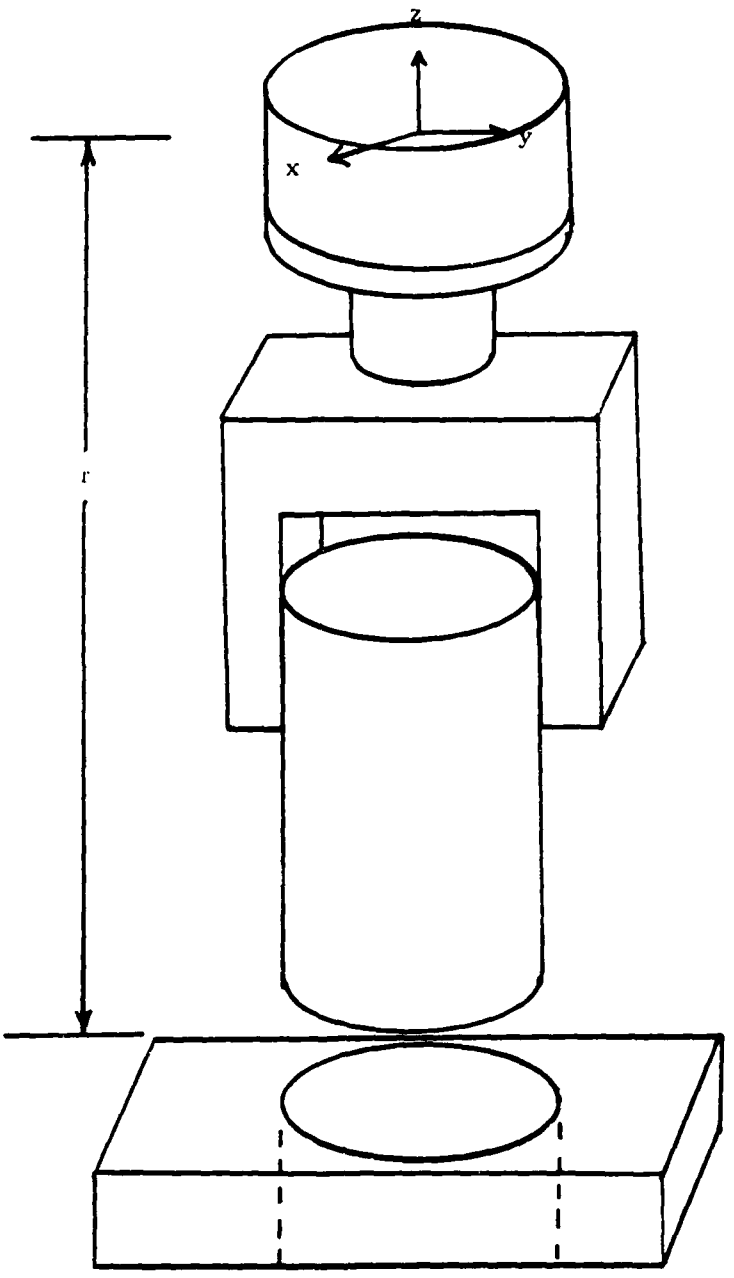
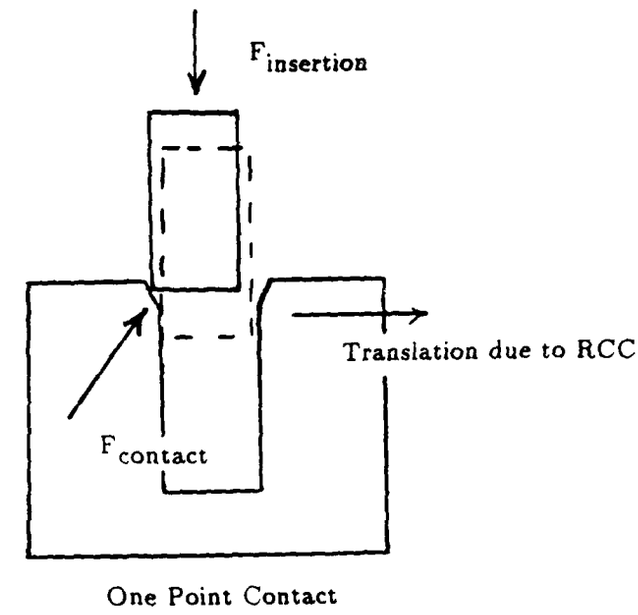
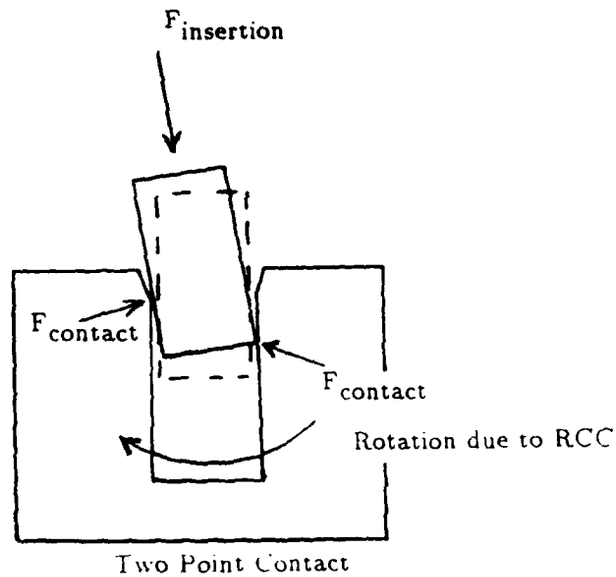


Figure 4.11

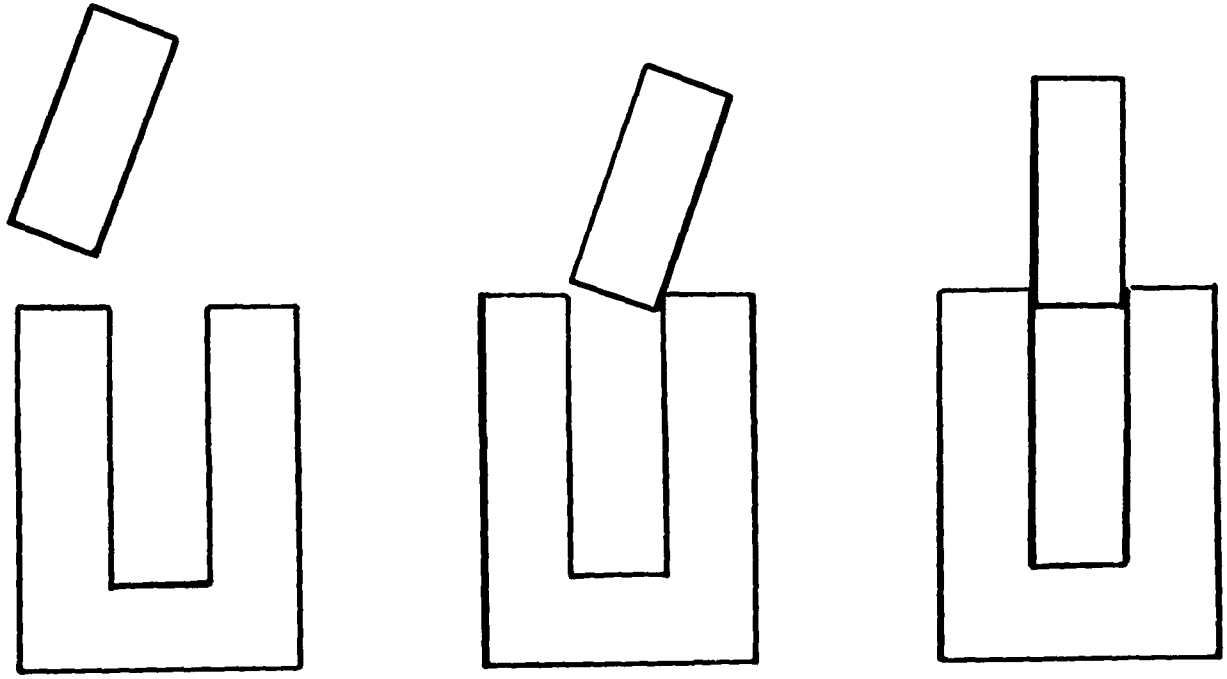


One Point Contact



Two Point Contact

Figure 4.12



Peg Tipped 10°

$W_{constraint}$ applied

Peg Orientation Righted

Figure 4.13

step, which is to rotate about the tool tip using interpolated motion, until the peg is vertical. A reciprocal wrench is simultaneously applied during this step, which keeps the peg in single point contact with the lip of the hole. Once the peg is aligned with the hole, the insertion procedes in the same way as for the chamfered hole. The constraint wrench for insertion is relatively simple, namely zero for those components associated with the tangible constraints of the workpiece. Choice of origin dictates the actual formulation. Using the origin shown in Figure 4.11, the following constraint wrench,

$$W_{constraint} = \begin{bmatrix} 0 \\ f_y \\ * \\ rf_y \\ 0 \\ * \end{bmatrix} \tag{4.8}$$

was used to maintain single point contact during the tipping strategy , while

$$W_{constraint} = \begin{bmatrix} 0 \\ 0 \\ * \\ 0 \\ 0 \\ * \end{bmatrix} \tag{4.9}$$

was used to complete the insertion for either case. The "*" entries indicate unspecified terms, such as the f_z and w_z terms, since they represent elements of the impulsive

wrench space. They are monitored, however, in order to detect jamming or bottoming. The f_z term, in particular, must be closely monitored, as the RCC device is intentionally very stiff along this axis. This is discussed in Chapter 5.

Another important type of problem is contour following, as shown in Figure 4.14. Here a roller is to follow a contour, such as a template, which may or may not be well defined. Placing the origin in the sensor, whose location is well known, the task constraints can be more easily formulated. The constraint wrench for the coordinate system shown is:

$$W_{constraint} = \begin{bmatrix} * \\ * \\ f_z \\ * \\ -rf_z \\ * \end{bmatrix} \tag{4.10}$$

where, here again, the unspecified terms are monitored for threshold values which indicate the possible need for operator intervention. This single roller configuration is generally inadequate, however, for tasks where the tool's orientation must also be accurately controlled.

De Schutter and Van Brussel [1988] suggest two alternatives for detecting orientation error based on either velocities or forces. However, as they themselves noted, both methods have serious drawbacks. Tracking based on velocities is very sensitive to flexibility, since minor corrections in force may generate displacement errors which are interpreted as orientation errors. Tracking based on forces suffers from the presence of

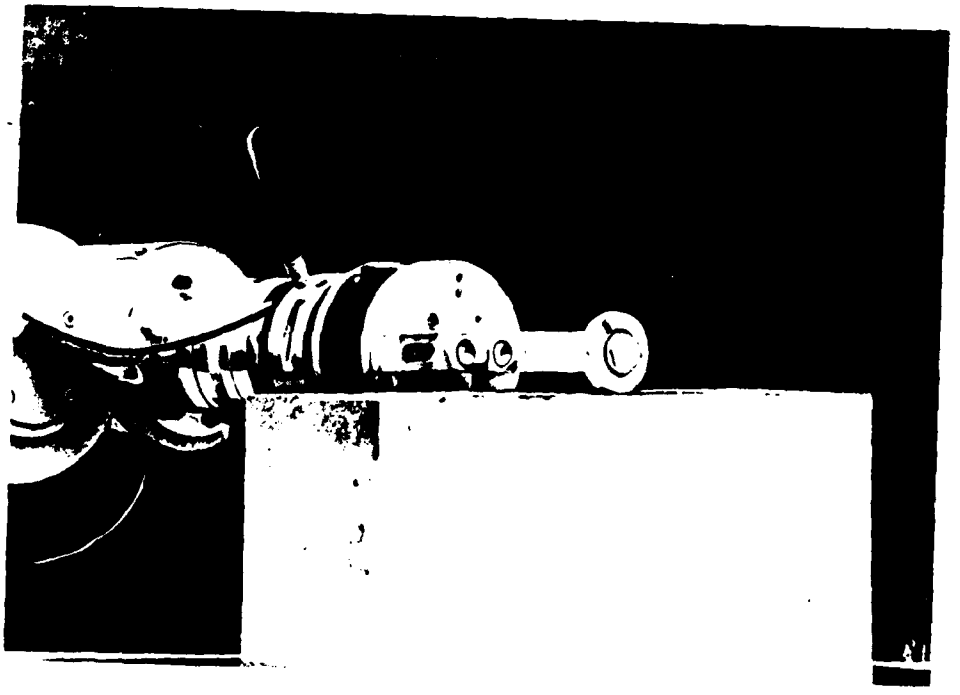


Figure 4.14

impulsive wrench components, especially due to friction, which then can be misinterpreted as orientation errors. A completely different approach was introduced and employed here.

Using a double wheel roller as shown in Figure 4.15, with the origin located in the sensor as before, the constraint wrench was specified as

$$W_{constraint} = \begin{bmatrix} * \\ * \\ f_z \\ m_x \\ -rf_z \\ * \end{bmatrix} \tag{4.11}$$

A tool could then be rigidly attached to this roller. For a tool orientation normal to the template surface, m_x was set equal to zero. An arbitray orientation angle to the surface could be specified by merely giving m_x an non-zero value. For example, with $m_x = 300$ torque units, the orientation angle was 10° to the template surface, as shown in Figure 4.16. This orientation could be maintained for tangential speeds up to about 200 mm per second on a flat surface.

These examples, along with the corresponding industrial tasks, are discussed in detail in Chapter 6.

4.5 Results and Conclusions

This chapter has focused on three conceptual issues necessary to design the controller. Specifically, these were the control architecture, the compensator design and the constraint formulation process.

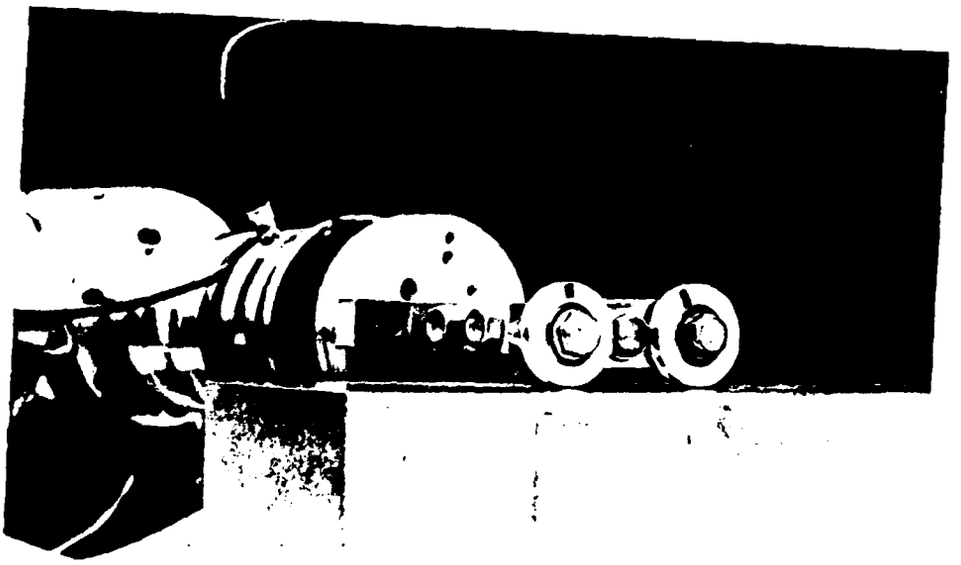


Figure 4.15

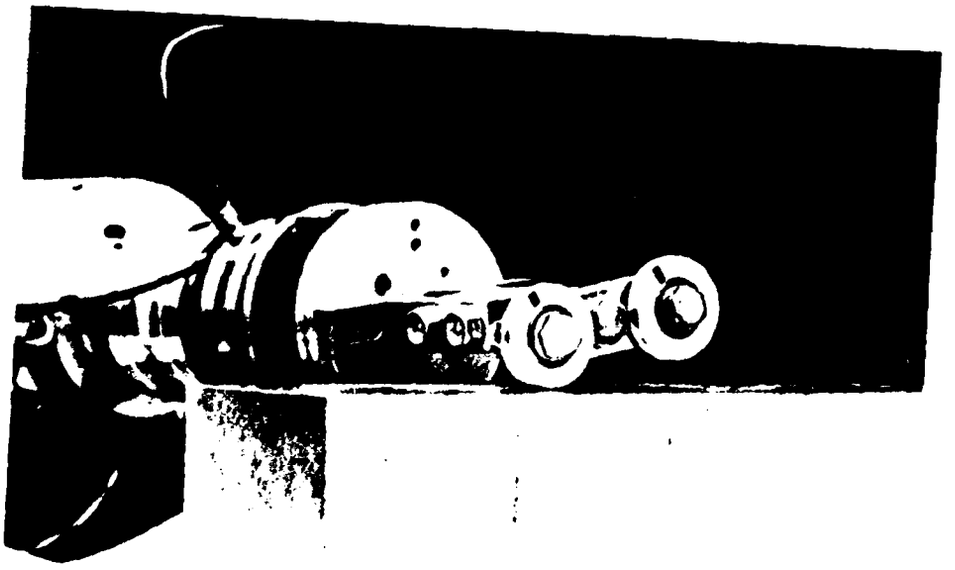


Figure 4.16

The architecture was predicated on the need for a practical method to augment an industrial robot with reciprocal wrench control. Issues such as coordinate systems, sample rates and disturbance rejection were presented, as were the reasons for using a split rate control structure. The resulting architecture was simple, easy to implement and robust in implementation.

The compensator design process was discussed and demonstrated by way of example. Tuning the servo, which is task independent because of the control architecture, is straight forward, and thus easy to duplicate.

Finally, a formalism for constraint formulation, based on the use of reciprocal screws, was outlined. Several sample tasks were described. Details of the implementation of these sample tasks, as well as performance results, are presented in Chapter 6.

CHAPTER 5
SYSTEM DEVELOPMENT

5.1 Introduction and Objective

Chapter 5 continues the design methodology begun in Chapter 4 by outlining the hardware and software development issues, as well as a description of the system integration and testing procedure. This discussion of hardware development begins with an examination of the wrench sensor system.

The cross-coordinated control strategy relies on a task space wrench sensor to provide real-time force feedback for closed-loop control. One important aspect of this design is the need for engineered, mechanical compliance, and how it was incorporated into the wrench sensor package. The configuration and integration of the transducer is also examined. The result is a versatile sensor system, readily applicable to both assembly and process tasks, as demonstrated in Chapter 6.

The chapter then continues with an examination of the hardware and software problems encountered in implementing the external, digital control loops described in Chapter 4. These hardware issues included input/output (I/O) handling, digital to analog (D/A) and analog to digital (A/D) signal conversion, as well as supplemental circuit requirements. Software issues included the choice of operating system, languages and programming requirements.

The chapter concludes with an examination of system integration and testing. An unexpected phenomenon was encountered during system testing, namely the self-

excited oscillation, or limit cycle. The source of this problem, together with several alternative solutions to it, are presented.

5.2 Wrench Sensor System

The first issue discussed is the need for, and physical realization of, engineered, mechanical compliance. This requirement was briefly discussed in Chapter 2, and is further developed here.

Several possible wrench transducer configurations are considered. They are discussed, together with details of the final choice. The system integration of the resulting transducer package is then discussed.

An important feature of this wrench sensor system is that both of the principal components, the RCC device and the wrist mounted force/torque sensor, are commercially available (off-the-shelf) items. As with the rest of this project, every effort was made to ensure that this research could be readily applied by others, with a minimal need for local fabrication, or the use of prototypes.

5.2.1 Engineered Passive Compliance

As already discussed in Chapter 2, the question associated with force control is not whether to use compliance, but where to use it. It has been demonstrated [Whitney and Nevins, 1978] that compliance can substantially increase the economical application of robots, through the ability to adapt to relatively unstructured environments, hence reducing fixturing costs. This capability is extremely valuable in justifying the cost of robot installation.

Studies performed by the GE Research and Development Center, as reported by Brownell [1988], have revealed that two-thirds of the cost of any robot installation was due to tooling, fixturing and applications programming. The high cost is largely

attributable to the need for a highly structured environment. The typical lack of sensors employed by industrial robots necessitates carefully planned movements, while maintaining a high degree of certainty about the position and orientation of all objects in the workspace. Compliance, on the other hand, can eliminate the need for those last increments of precision, which are always costly and may even prove to be technically impractical.

An equally important issue concerns the transition from pure trajectory control, where the end-effector is unconstrained by the environment, to hybrid twist and wrench control, where the end-effector is partially constrained. Paul [1987] reported that the most difficult part of the design problem with Scheinman's "Maltese Cross" wrist force sensor was not so much the transducer itself, but rather the design of a force overload mechanism to prevent it from being damaged on impact with a rigid environment. With a workbench stiffness of 8000 lbs/in, forces can build up at a rate of 96 lbs/msec at end-effector speeds as low as 1 ft/sec [Brownell, 1988]. Obviously this force build up must be accomodated. The problem is bandwidth.

On impact, a force sensor experiences significant force interaction within a few microseconds. Paul [1987] observed that force transducer signals are processed by regulators with well defined, minimum delays on the order of milliseconds. This means that contact damage will most likely have already occurred before the regulator can respond. Efforts to overcome this characteristic limitation of "active compliance" designs have been the focus of an ongoing research effort at General Electric [Brownell, 1988], but instabilities attributed to the non-colocation of force sensor and actuators have not yet been overcome.

There are several ways that mechanical compliance may be embodied in a system. It may be inherent in the component parts, such as the links, actuators and drive trains, or it may be intentionally introduced through tooling and fixtures. Rivin [1988] presents techniques for measuring and analyzing the compliance distribution in a manipulator structure, including both the structural elements and the energy transforming devices. A detailed methodology for evaluating the mechanical compliance of manipulators containing parallel mechanisms, like the GE P60, were reported by Leu et al. [1985].

These experiments have shown that most of the compliance in a modern industrial robot is attributable to the drive train. This is due to the use of heavy, rigid links for end-effector positional certainty as discussed earlier. While electric motors, themselves, do exhibit significant compliance, the effect is greatly reduced by the square of a usually high transmission ratio. Hence, the inherent manipulator compliance is relatively small for industrial robots, like the GE P60. One should recognize that this stiffness is intentional, since these manipulators were designed to operate with the end-effector unconstrained. Under pure trajectory control, uncompensated compliance serves only to degrade performance.

For example, this inherent compliance in the drives contributes to uncontrolled deflections in the end-effector tip position, reducing positional accuracy. Furthermore, excessive compliance will tend to reduce the servo control bandwidth. The optimal solution for manipulators that must also be able to perform contact tasks is to introduce a highly structured, anisotropic, mechanically compliant device in order to satisfy the need for compliance along certain axes, without sacrificing positional accuracy along the

others. This well defined compliance will dominate the total manipulator compliance of a well designed industrial robot.

The Remote Center Compliance (RCC) device developed at Draper Laboratory [Whitney and Nevins, 1978], is just such a device. A model RC-212-RS RCC device, manufactured by the Lord Corporation, is shown in Figure 5.1. This design makes use of elastomeric shear pads. These shear pads are typically composed of alternating layers of elastomer (rubber) and rigid washers. While pad characteristics vary, the axial stiffness, as compared to the shear stiffness, may be several hundred to one [Whitney and Rourke, 1986]. This anisotropic behavior is what makes these devices so useful. A lateral force applied to a part that is connected to the bottom plate will tend to cause only a lateral translation, as shown in Figure 5.2, with otherwise negligible re-orientation. Of course, these devices exhibit this ideal behavior for only a small range of motion, of the order of $\pm$ 0.25 inch, but this is usually sufficient to accomplish a typical industrial task. A limited amount of rotation flexibility, in response to an applied couple, is also characteristic of the RCC device. The actual rotation, which ideally is about some projected center of compliance, is less predictable. A much improved design, which makes use of a push-pull configuration of the shear pads, is under development [Whitney and Rouke, 1986].

The application of these devices to assembly, particulary insertion tasks, is well known [Whitney, 1982]. When used with a chamfer, they are highly effective at ensuring tip insertion, as discussed in Chapter 4. In fact, with sufficient hole clearance, the RCC device is capable of ensuring a successful peg insertion, without the need for any additional active accomodation as provided by wrench control.

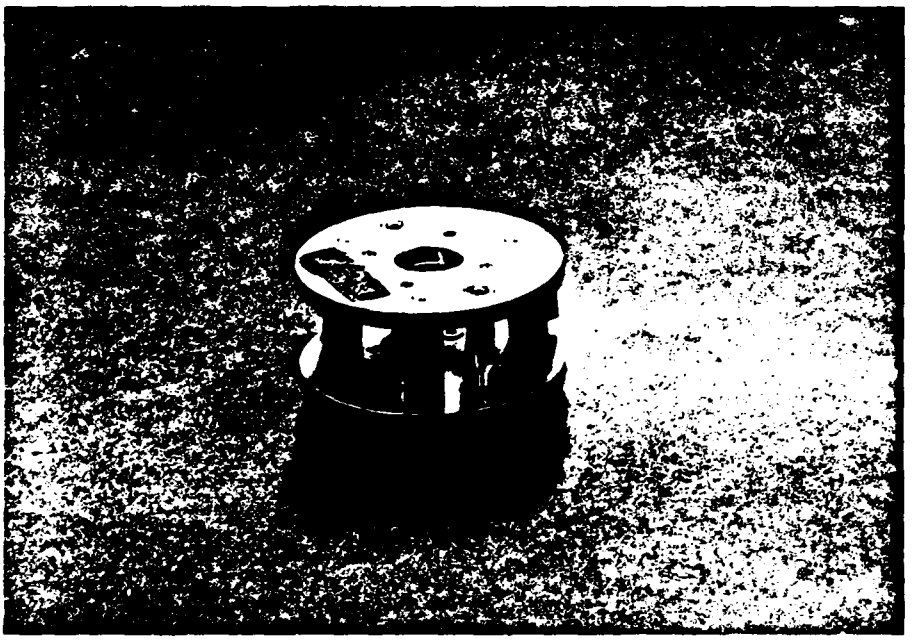


Figure 5.1

RCC DEVICE

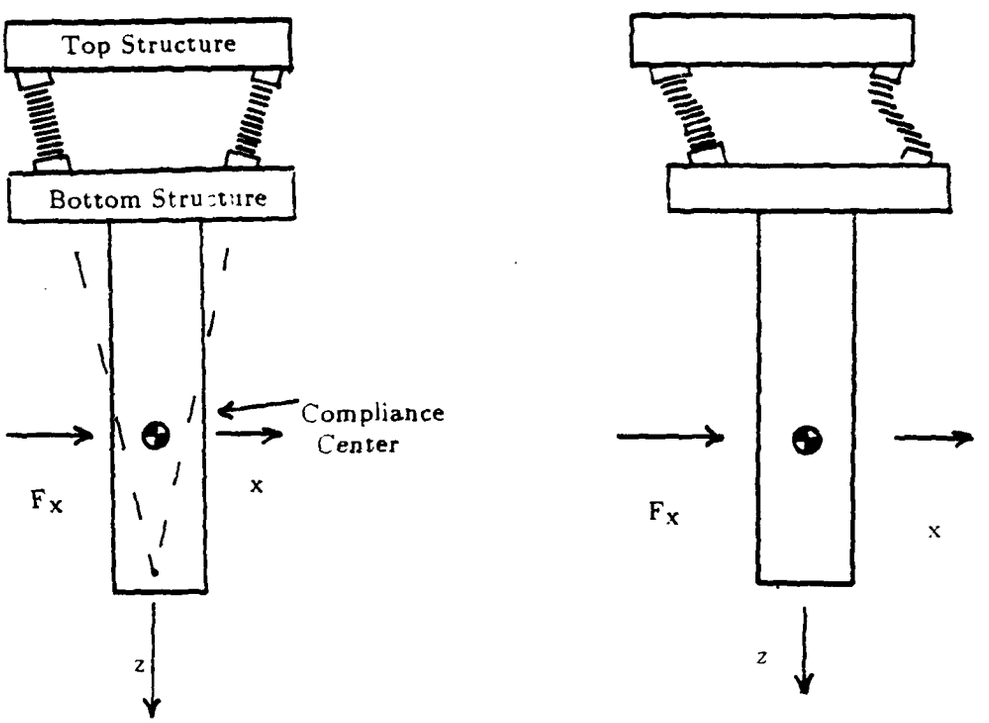


Figure 5.2

As part clearances are reduced, however, the percentage of assembly failures increases dramatically. Spending more money on improving robot repeatability, tooling and fixturing can reduce the positional uncertainty. However, for so-called precision assembly tasks (clearances on the order of 0.001 inch or less), such an approach is usually not economically feasible. Cost effective precision assembly can be achieved, however, by combining engineered compliance with force sensing.

In 1979, Draper Laboratory developed the Instrumented Remote Center Compliance device, or IRCC. This device exhibits the advantages of both passive and active compliance. The passive mechanical compliance acts as a high (infinite) bandwidth error absorber, able to safely accomodate small positioning errors beyond the bandwith of the force control loop. The instrumentation, which in the latest model is an array of optoelectronic position sensors [Seltzer, 1986], provides the sensor feedback necessary to modify the robot position command, and thus accomodate the build up of positioning errors. This combination facilitates the use of the more complex algorithms, necessary for precision assembly. Unfortunately, no commercial version of the IRCC is available for purchase, although Draper Laboratory has offered to build one for the author at a cost of $10,000. Instead, an alternative but functionally similiar device, composed of commercially available components, was used. This alternative proved extremely effective for both assembly and contact process applications.

5.2.2 Wrench Sensor

In order to actively control the constraint wrench applied by the end-effector, some form of force measurement was required. Historically, three different transducer configurations have been employed.

The first of these is the pedestal force sensor. Here, the workpiece platform itself is instrumented. The main advantage to this approach is that dynamic loading of the end-effector has no effect on task performance. Unfortunately, this approach is task dependent. As such, it does not facilitate the exploitation of the flexibility usually necessary to justify the added expense of a robot installation instead of fixed automation.

The second approach involves the use of joint torque sensing, usually by measuring the current at each motor. Strain gauges on the output shaft have also been suggested. This approach has great pedagogical appeal, and periodically reappears in the literature [Asada and Lim, 1985], characteristically including the results of extensive simulation. The problem, however, is that this approach requires an accurate, detailed, *real-time knowledge of all dynamic, gravitational,* and frictional loads in order to reconstruct the applied wrench. This requirement has effectively precluded a successful demonstration of this type of configuration for industrial applications.

Certainly the most common transducer configuration is the wrist mounted force/torque sensor. These devices, which typically use an array of strain gauges, are located close to the end-effector. As such, they are far superior to joint torque sensors in measuring applied wrenches, while still retaining the necessary flexibility of application not possible with pedestal force sensors. Hence, a wrist mounted wrench sensor was chosen for this research, specifically the Lord Corporation model 15/50, as shown in Figure 5.3.

5.2.3 Compliant Wrench Sensor System

As previously noted, the IRCC device, which combines the engineered passive compliance of the RCC with the active compliance attributable to wrench sensing, is not

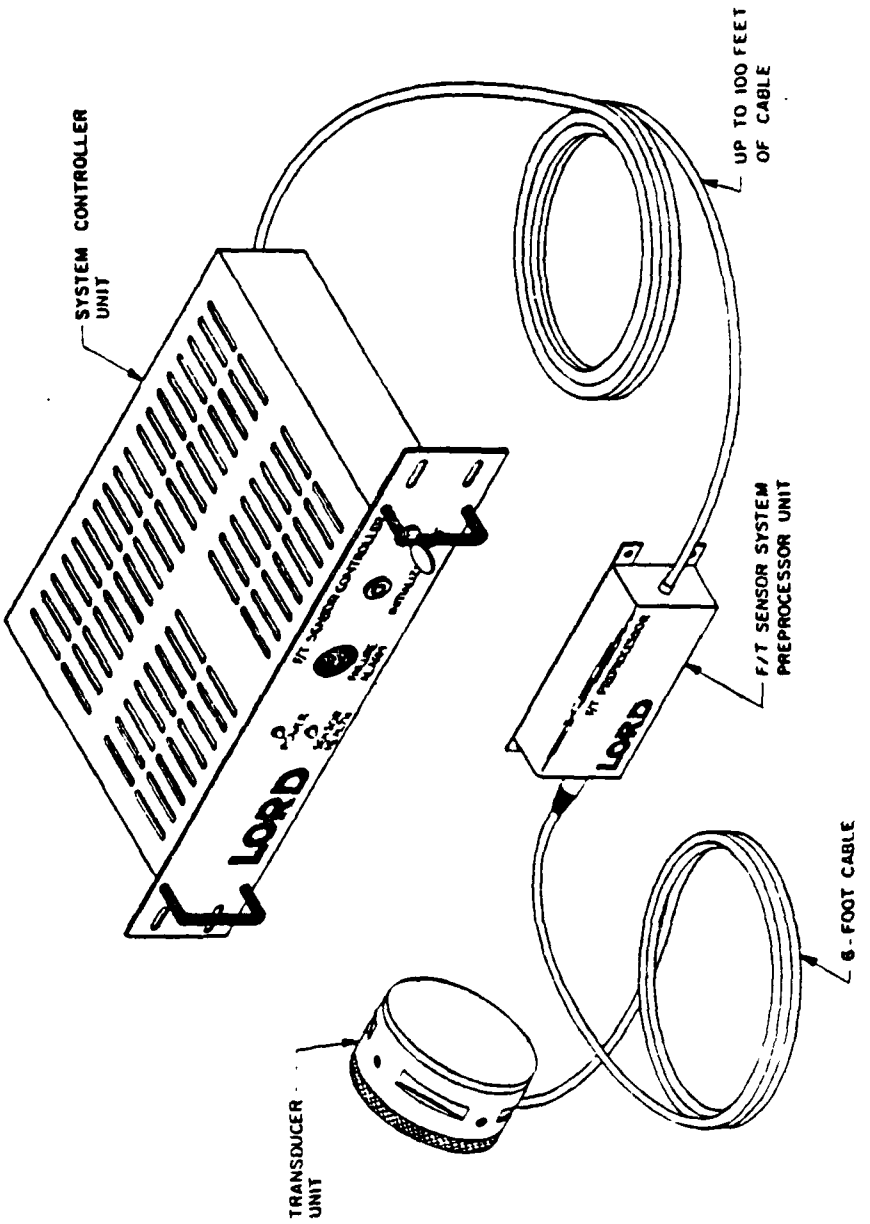


Figure 5.3 F/T 15/50 System

commercially available. Therefore, the author decided to combine the RCC device with a wrist mounted, force/torque sensor to create a compliant wrench sensor, as shown in Figure 5.4.

The force/torque sensor is rigidly mounted to the end-effector tool face. Since the wrench control loop is closed on this non-colocated sensor, it is desirable to minimize the dynamics between this sensor and the actuators, as discussed in Chapter 2. The applicability of this sensor package to assembly tasks is essentially the same as for the IRCC, as described by Seltzer [1986]. For process tasks, like grinding, cutting or deburring, the RCC served primarily to increase the compliance parallel to the constraint wrench, while the force/torque sensor was used to close the servo control loop. In addition to enhancing stability, the mechanical compliance served to increase the positional resolution of the robot along those compliant axes.

The robot's positional repeatability is approximately 0.005 inches. This meant that an incremental force command could only be resolved to about five ounces, based upon manipulator stiffness. With the addition of the passive compliance provided by the RCC device, the incremental force command resolution was improved to less than one half ounce, a ten-fold increase. Since grinding and deburring forces are typically on the order of ten ounces [Gillespie, 1987], this higher positional resolution provided the necessary positional resolution to successfully accomplish these tasks.

5.3 Digital Computer Control

In recent years, the digital computer has largely replaced the analog controller for many applications. Indeed, much of the success of industrial robots is directly attributable to recent advances in digital computing, particularly in the area of

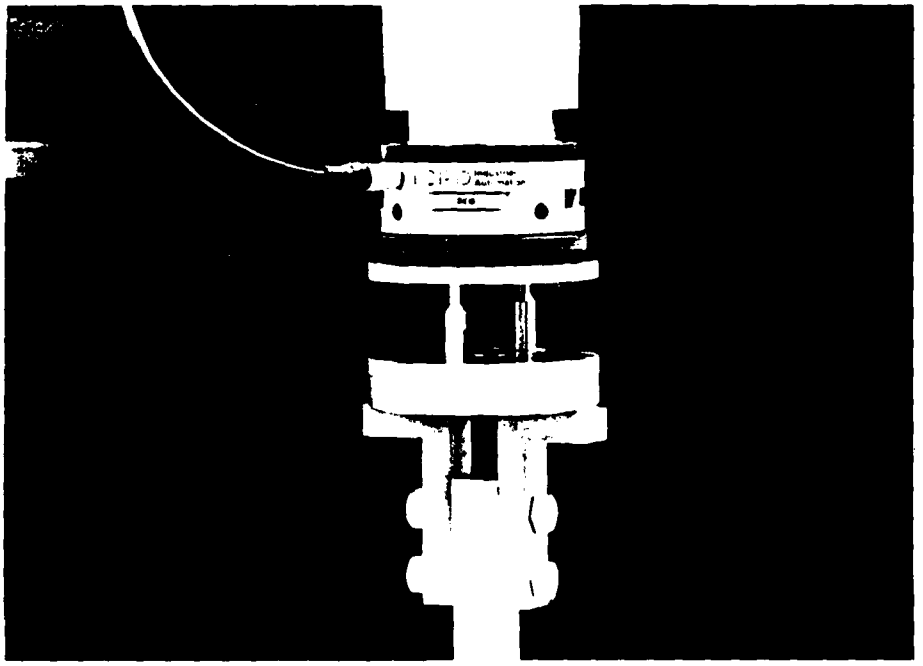


Figure 5.4

microprocessors. These small, and relatively low cost devices have brought the power of the digital computer to a wide range of applications, including the industrial robot.

Digital computer control has a number of advantages over its analog counterpart. Analog systems require constant adjustment, due to parameter drift with time, as well as that due to variations in temperature and even humidity. This problem can be partially circumvented by the use of fixed, plug-in, analog compensation networks. However, these run counter to one of the main factors justifying the expense of robot installation, namely flexibility. Furthermore, these networks preclude fine tuning to correct for minor system variations.

Another disadvantage of analog compensation is that only a single set of loop gains can be implemented. No facility exists for adaptation of feedback gains while the system is operating (on-the-fly). This can be a major drawback for robot manipulators whose loads may vary dramatically, particularly for high speed applications.

Digital control, on the other hand, does not suffer these limitations. The digital representation of data makes it immune to time, temperature and humidity variations. Furthermore, by implementing control loops in software, it is possible to alter control loop parameters during program operation, thus facilitating not only fine adjustment, but also the use of more "exotic" control strategies, as they become available. These digital systems are not, however, without some drawbacks.

In order to implement a control loop in software means employing a discrete time, or sampled data approach together with an associated sampling rate. If the sampling rate is too slow, a phenomenon known as aliasing can develop. This undesirable phenomenon can be avoided, however, if the sampling rate is at least twice the control bandwidth, and preferrably at least 5 to 10 times this value. The closed-loop

position bandwidth for the servos found on the GE P60 averaged about 2 Hz, based on the usual definition (frequency at which the output magnitude is reduced by 3 Db from DC value), as can be seen in Figure 5.5. The force control loop, on the other hand, operated at 104 Hz.

At this sampling rate, the sampled data system can perform as well as the equivalent continuous system. Actually, it can perform better, as pointed out by Åström and Wittenmark [1984]. Besides emulating the conventional control strategies of analog compensation, many additional options, unique to digital control, are available. Deadbeat control, in particular, fits this description.

The other drawback of digital control is the need to interface the digital computer's discrete time output with the plant, which is typically analog. This may generate the need to build some additional circuitry, as was the case here. This circuitry is described in the next section.

5.3.1 Hardware Development

The most important piece of hardware for digital control is the digital computer, itself. The author used a DEC PDP-11/23 industrial minicomputer. This computer system had three main advantages:

1) It had already had real time operating system, RT-11 V4.0, installed. This operating system is discussed in the next section on software.

2) It had a sufficient number number of A/D and D/A channels already installed, as well as several 16 bit parallel comunications ports.

3) Most important to any actual laboratory application, it was available.

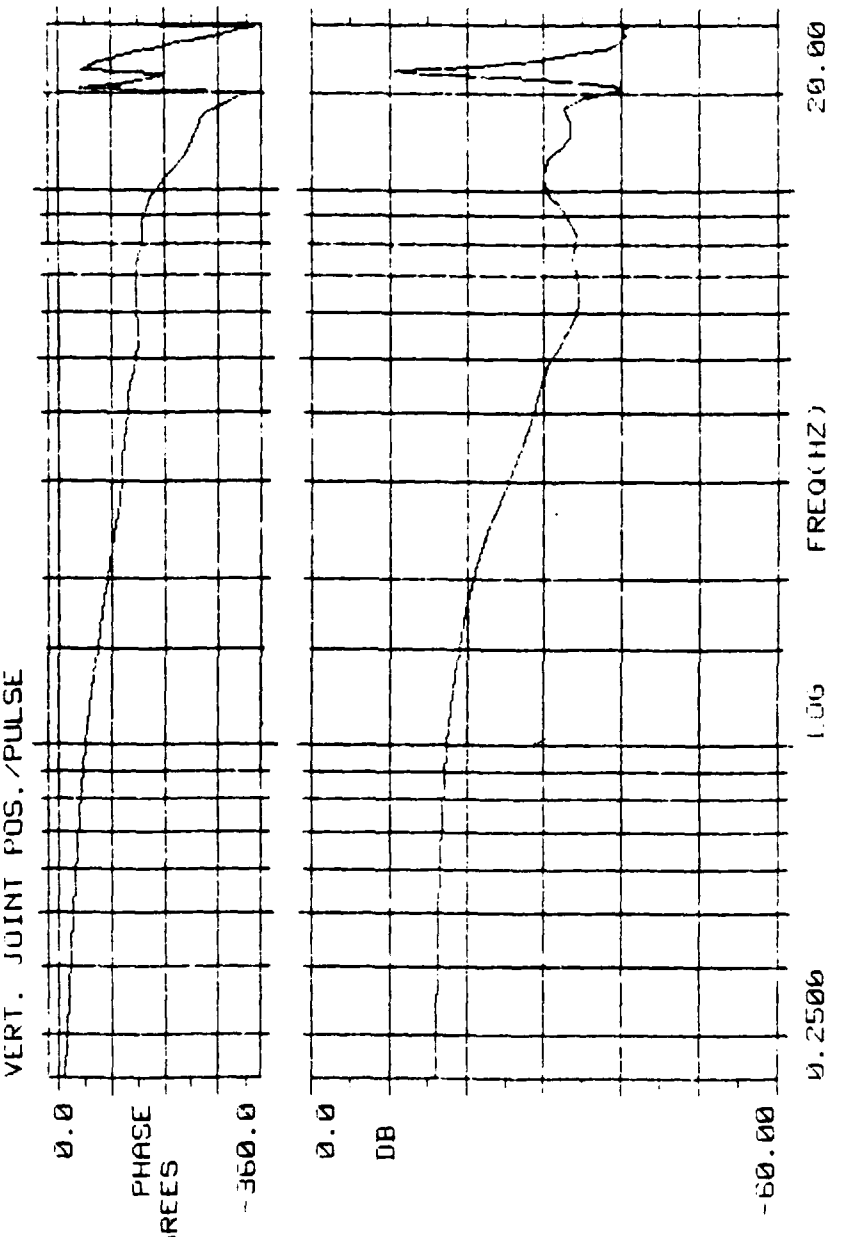

VERT. JOINT POS./PULSE
0.0
PHASE
DEGREES
-360.0
0.0
DB
-60.00
0.2500
1.00
FREQ(HZ)
20.00

Figure 5.5

While certainly not a new system, the DEC PDP-11/23 has a demonstrated record of reliability in industrial applications. Several other factors should also be considered when choosing a system if, indeed, several candidate machines are available.

The first factor is computational capability. Does the machine have sufficient throughput capability for the computational burden? If so, does it offer flexibility and expandability, in case new requirements are generated? Is there a real-time operating system and support for high-level languages on the candidate machine? What about peripherals? Can the system satisfactorially interface with other equipment? Fortunately, the answer to each of these questions was yes, for the PDP-11/23.

The only custom built hardware required for this design was a series of standard summing circuits, as depicted in Figure 5.6. These circuits scaled and combined the the force control signal, a voltage, with the velocity command signal generated with the RC 1560, also a voltage, for each axis. The only significant design limitation on the supplemental wrench signal was the requirement that its magnitude could not exceed one volt. This limitation is one of the safety features built into the RC 1560, to prevent a run away actuator, and no effort was made to defeat it. None was needed. A one volt signal, if applied simultaneously to all actuators, could move the end-effector as much as 8 inches. This far exceeded the motion needed to implement a useful wrench control augmentation, since assembly clearances, for example, are often times measured in thousandths of an inch.

5.3.2 Software Development

As the hardware and software issues of the conceptual design began to partition themselves, it was possible to begin addressing the problem of software development. There are a number of factors affecting this process. Foremost among these is the

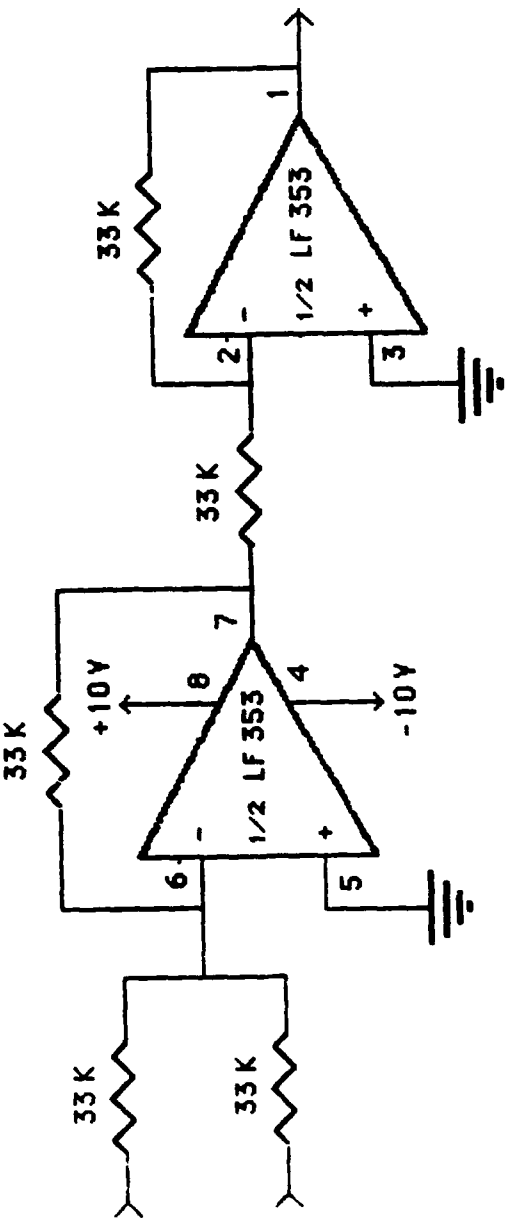


Figure 5.6 Summing Circuit

operating system, which, after all, acts as the basis for program development as well as being the intermediary between the user, the system hardware, and external devices.

While far from new, the RT-11 V4.0 operating system for the PDP-11/23 is a real-time operating system. Lawrence and Mauch [1987] define real-time computer systems as those whose "temporal performance" is critical to the system it controls. In the case of the RT-11 system, this meant that a vector of wrench data was received, processed, and a correction signal generated and dispatched, before the next input was received. This all occurred at 104 Hz. One should recall that many chemical process applications measure time constants in hours, as compared to milliseconds for industrial robots and similar automation systems. Another issue is programming languages.

Computer programming languages vary from low-level machine languages to applications-oriented, high-level languages. Thus the level of language, as well as the specific language choice, are required. Obviously different languages will be more suitable to different types of tasks, but these differences should not be overstated. For this application, there were three requirements to be addressed:

1) The language or languages used had to be capable of bit manipulation. That is, they had to permit the addressing of the registers, directly, in order to facilitate the interface of external hardware.

2) The language needed to be capable of both floating point and integer computation in order to support both the DRV11 D/A board and the PID algorithm.

3) Most important, it had to be available! That is, the candidate assembler, compiler or interpreter which was compatible with the PDP-11/23 minicomputer, had to be reasonably available.

Unfortunately, it is often difficult to obtain accurate information about which languages and compilers will produce the most efficient code. While benchmark programs are available, their applicability to a particular task may be questionable. Fortunately, this decision was simplified by the fact that only two languages capable of the low-level bit manipulation required were available for this machine, an assembly language, MACRO-11 V4.0, and an early version of C, which was not a DEC product, by the way.

C is a language developed at Bell Laboratories in the mid-1970's, primarily for systems programming. It is extremely powerful, and has been the language of choice for graphics programming at our facility. Unfortunately, the C compiler that was available for the PDP-11/23, a fairly old machine itself, was a very early version, with poor I/O handling and somewhat eratic compilation. This annoying characteristic is dangerous for computer control. Furthermore, due to the age of the computer system, no newer version of a C compiler was reasonably available. This left MACRO-11 as the only alternative for board level programming.

This situation, however, was hardly catastrophic. Despite the advantages of higher level languages, users often times find it necessary to write at least part of their code in assembly language, because of its inherent speed and bit manipulation capabilities. That was the case here. MACRO-11 was used to interface the PDP-11/23 with both the I/O boards and the force/torque sensor, while FORTRAN was used to code the actual control programs.

Unfortunately, again due to the age of the machine, this was an old version of FORTRAN, specifically FORTRAN IV V2.5. Fortunately, this compliler was consistent and reliable. An old adage says that good code in a poor language is always better than

poor code in any language. Apparently the code was good enough. Furthermore, our success clearly demonstrated that the results of this research did not depend on some state-of-the-art computing or programming capability, unavailable to the average industrial robot user.

FORTRAN was developed in the mid-1950's as an applications language, largely to replace assembly language, which is cumbersome to work with on most machines. Though FORTRAN has gone through much evolution, most engineers are at least familiar with it. Furthermore, it remains the dominant language of scientific and engineering applications.

No actual code has been included here. First, assembly language routines are always hardware specific and, as such, would be useless to anyone without exactly the same hardware. Second, the control programs were typically coded in one hundred lines or less, consisting of subroutine calls (to the assembly language routines) and the PID algorithm, which was presented in detail in Chapter 4. The author used no special tricks (nor did he know any) to make his code perform better than similiar code that would be generated by any other mechanical engineer. Those seeking samples of this type of code should refer any of the excellent programming references available, as did the author.

5.4 System Integration and Testing

While the top-down approach is appropriate for the design process, exactly the opposite approach must be used in order to integrate, test and, if necessary, debug the system. This bottom-up approach means that testing begins at the lowest possible level of partitioning. Experience has shown that by designing from the top-down, and testing

from the bottom-up, very substantial savings in time and effort can be realized [Johnson, 1987].

For hardware, this means testing boards, transducers, and especially any custom built circuits. The time spent validating these elements is time well spent, especially when that bad op-amp or broken jumper wire is discovered before it can damage the other equipment, or consume hours of debugging efforts after the system is fully assembled.

The same philosophy must also be applied to the software. With the overall programming structure already determined, the software is typically partioned into a number of computational subroutines and device drivers. Each of these should be tested exhaustively, through the use of driver programs. These are simple calling programs that do nothing more than provide test inputs, and a convenient means of examining the corresponding outputs. Device drivers are similiarly tested with driver programs, but auxiliary test equipment, like oscilloscopes or frequency analyzers, may also be required to examine output response. By testing these software modules in isolation, they can later be combined into large software packages with relative confidence.

After verifying the software and hardware components, system integration and testing can procede. Because of the speed inherent (and essential) to digital computer control, the testing at this level may require that operations be slowed for observation. This can be accomplished by the use of break points in the program, so that the system state can be compared with that expected. One might note that commercial robot controllers, like the RC 1560, have the ability to single step through a program built in, thus facilitating similiar testing of path programs.

The industrial workplace requires that the system's vulnerability to its environment be minimized. This is due, in part, to the enormous range of dynamic parameters characteristic of industrial settings. Kilowatt welding torches may share the environment with microwatt control circuitry. Hence, every effort to preclude interference must be made. Correspondingly large mechanical effects, such as robot arm inertia, must also be insulated from the more fragile mechanisms like transducers, workpieces and human operators!

Fortunately, much of the adverse electrical interference can be moderated. Shielded cables can mimimize capacitive pickup between conductors. The use of an earth ground, to prevent ground loop effects between electronic assemblies, may also prove beneficial. The use of separate power supplies for the controller and other power consuming system components, such as large motors or plasma torches, is also recommended. If not possible, then power conditioning through the use of transformers or filters may help.

The key is to remember that the environment will provide a variety of obstacles to system implementation. A bottom-up structure of system integration and testing will greatly enhance one's ability to isolate and correct the inevitable problems that occur.

One such noteworthy, and totally unexpected problem occured during the testing of the individual axis compensators. With the end-effector in contact with a rigid work piece, a step command was issued by the PDP-11/23, in order to verify compensator performance. The response was essentially as predicted in simulation, except that shortly after the output reached its steady state value, a limit cycle, as shown in Figure 5.7, was detected.

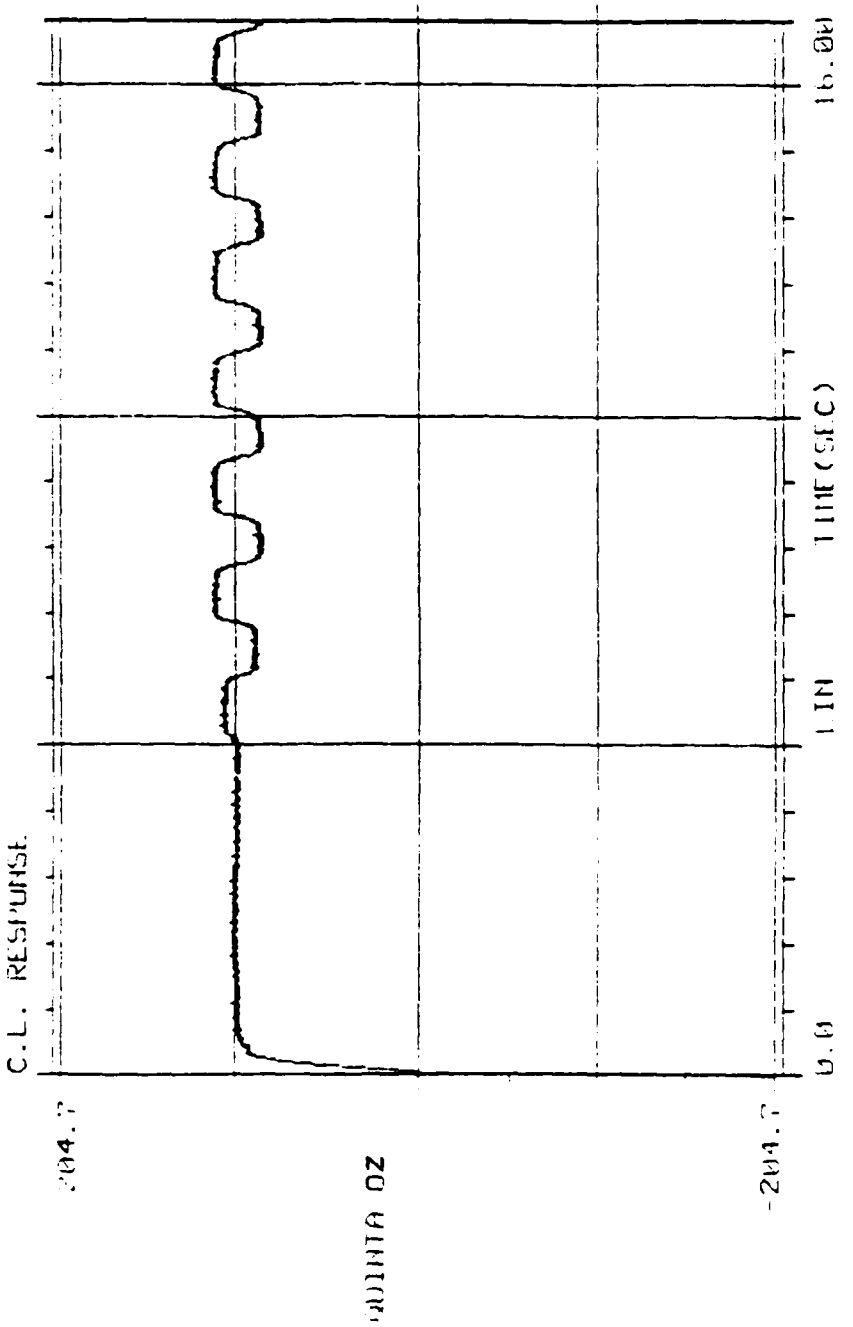


Figure 5.7

De Schutter [1986] reported a similiar phenomenon, without further comment. This was not unreasonable, however, since some limit cycle behavior is characteristic of hydraulic actuators [Merrit, 1967], such as those found on the T3-556 robot used by De Schutter [1986].

Such phenomenon was certainly not expected, however, for a dc servomotor actuated robot. No such oscillations were reported by Stepien et al. [1987], who used GE P50 robot and an external force control loop. However, they did operate at a very low loop gain, namely unity. When the author also lowered the compensator gain to unity, the onset of limit cycling was delayed by as much as 15-20 seconds, which probably explains why this type of result was not reported by Stepien et al. [1987].

De Schutter and Van Brussel [1988] did address this issue in a later paper. They concluded that this self-excited oscillation was due to the limited positional resolution of an industrial robot, and was therefore characteristic of force control in a static situation. While this may be true for the T3-556, experiments with the GE P60 suggested otherwise.

First, this phenomenon also occured with the P60 in motion, specifically contour tracking. Interestingly, the better the taught program traced the required trajectory, and hence the smaller the error signal from the force controller, the sooner the limit cycle occurred. Furthermore, these oscillations were far more pronounced on the bend axis than the vertical or horizontal axis. This seemed to further contradict the position resolution hypothesis of De Schutter and Van Brussel [1988], since the bend axis has a positional resolution (as seen at the end-effector) on the order of five times better than that of the horizontal or vertical axis. Clearly, some cause other than positional resolution was responsible, and this cause appeared to be friction.

In his discussion of limit cycles in non-linear systems, Ogata [1970] presents the Van der Pol equation:

$$m\ddot{x} - f(1-x^2)\dot{x} + kx = 0 \qquad (5.1)$$

where m, f and k are all positive quantities. This equation is non-linear in the damping term. One can easily see in this equation that as x gets smaller, the sign on the damping term becomes negative. Physically, this corresponds to putting energy back into the system. For large values of x, this term is positive, and represents the removal of energy (energy dissipation) by the damping term, as one might expect. This appears to correspond precisely to what was observed during our testing.

As long as the wrench error signal was sufficiently large, *no limit cycle behavior* was observed. Furthermore, while loop gain obviously has no effect on positional resolution, it did affect the onset point of the limit cycles. One can see how this effect may relate to the scalar multiplier, f, in Equation 5.1. Furthermore, the flat top and bottom of the limit cycle wave form is characteristic of friction induced backlash [Merritt, 1967]. Since the bend axis, with its harmonic drive and belt drive, could reasonably be expected to experience higher friction than the ball screw drives used on the vertical and horizontal axes [Rivin, 1988], the probable cause of the limit cycle would appear to have been identified. It remained to eliminate it.

Clearly, these oscillations were undesireable. They caused an obvious degradation in force control. Also, the oscillations suggest an increased wear on system drive components. Furthermore, while possibly acceptable for some process applications, particularly with a reduced loop gain to delay onset, this oscillation was clearly

unacceptable for precision assembly. The usual way to overcome this problem in hydraulic systems is to introduce a dither signal.

A dither signal is a characteristically high frequency oscillation which is intentionally added to the command signal. The backlash due to static friction is eliminated by keeping the servo system in continuous motion. This was attempted on the GE P60. with some success, as shown in Figure 5.8.

However, the frequency range required to produce a positive effect was much lower than expected, and was of the order of 5-10 Hz depending on the waveform and the amplitude of the dither signal. The governing factor seemed to be the area under the curve, corresponding to the power input to the motor. As long as sufficient power was drawn to keep the rotor in continous motion, the limit cycle oscillations were suppressed, regardless of whether a square wave, saw tooth wave, or sine wave was used as the dither signal.

Therefore, a square wave was chosen, since it was fairly easy to construct a timer circuit using integrated circuit (IC) components, as depicted in Figure 5.9. This circuit could generate the signal fairly unobtrusively. A relay could then be connected to enable the adding of the dither as required. This solution, however, was certainly not altogether satisfactory. For one thing, the dither frequency was present in the output, causing a force error of about 3-5%. Although superior to the 15-20% error characteristic of the limit cycle, such error was still undesirable. Of more concern, however, was the fact that this signal intentionally induced a continuous oscillation in the actuator, suggesting the same problem of premature wear of the drive components as with the limit cycle itself.

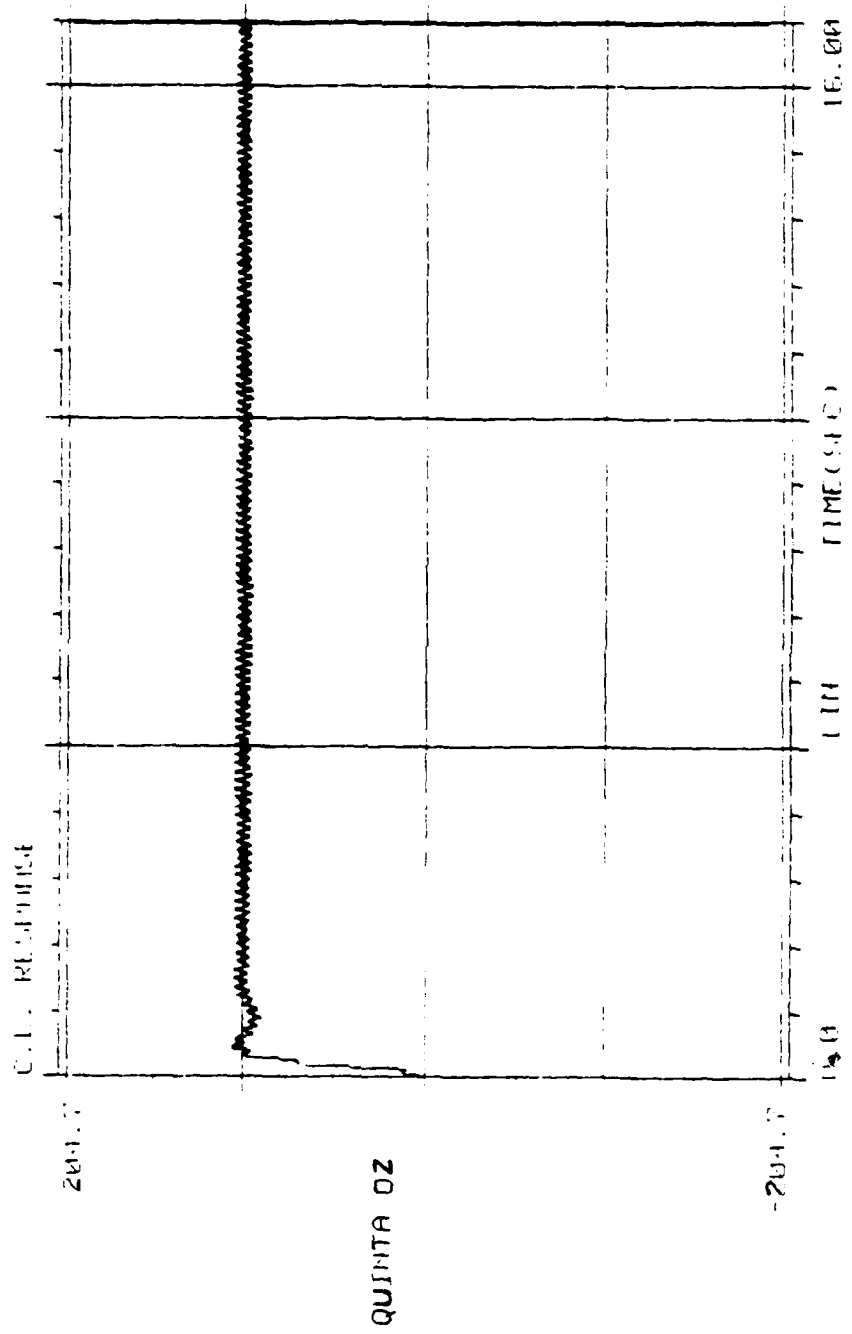


Figure 5.8

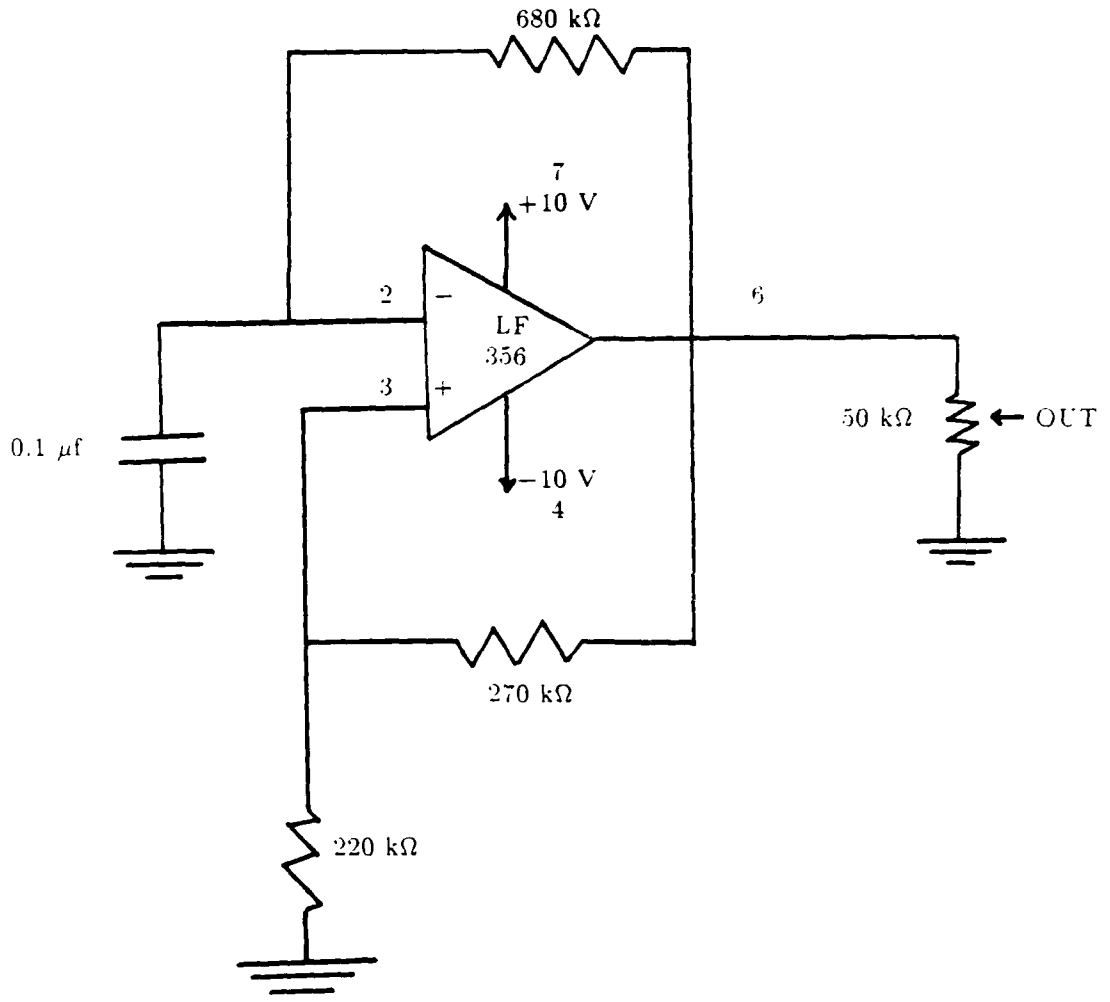


Figure 5.9

Another solution, although seemingly only useful for unilateral constraints, was the intentional introduction of a trajectory error. By intentionally teaching a path with "significant" positional error, the error correction signal was significant throughout program execution, as depicted in Figure 5.10. While successful in preventing the limit cycle oscillations, such an ad hoc solution certainly has a number of limitations. The very fact that the required positional error can be preprogrammed implies a fairly high level of a priori knowledge about task geometry, thus degrading the robustness of the implementation. More important, this approach was not suitable for bilateral constraint tasks, such as assembly. It did have the advantage that no additional oscillation was introduced into the drive system, nor was any additional hardware required. Fortunately, the final solution employed neither of these techniques.

An important feature of this control strategy was to be able to incorporate multiple actuators to control the constraint wrench, while they simultaneously generated the desired twist. When this last step of the system integration phase was tested, the limit cycle effect essentially disappeared! Youcef-Toumi and Ro [1986] had earlier examined the use of two motors to drive a single joint of a direct-drive arm. They found that this redundancy tended to bias away the effects of the non-linearities in the servos. Apparently by mapping the constraint wrench error into joint space through the use of $(J^*)^T$, and thus distributing the constraint load among multiple actuators, a similiar, desirable effect was observed. This can be seen in Figure 5.11, where the vertical, horizontal and bend axes each contributed to the constraint wrench.

The final, integrated system is shown in Figure 5.12. The major elements depicted include the compliant wrench sensor, consisting of an RCC device and a commercial force/torque sensor, the manipulator itself, the RC 1560 controller including the teach

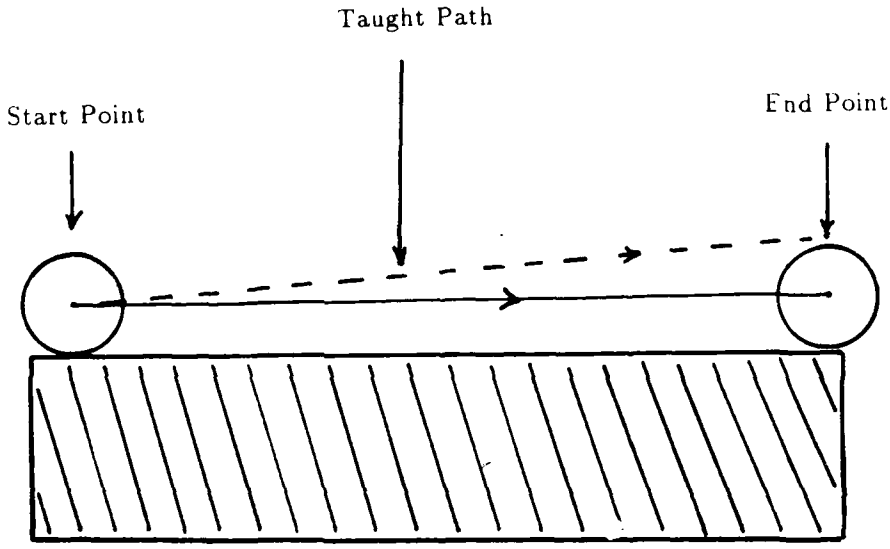


Figure 5.10

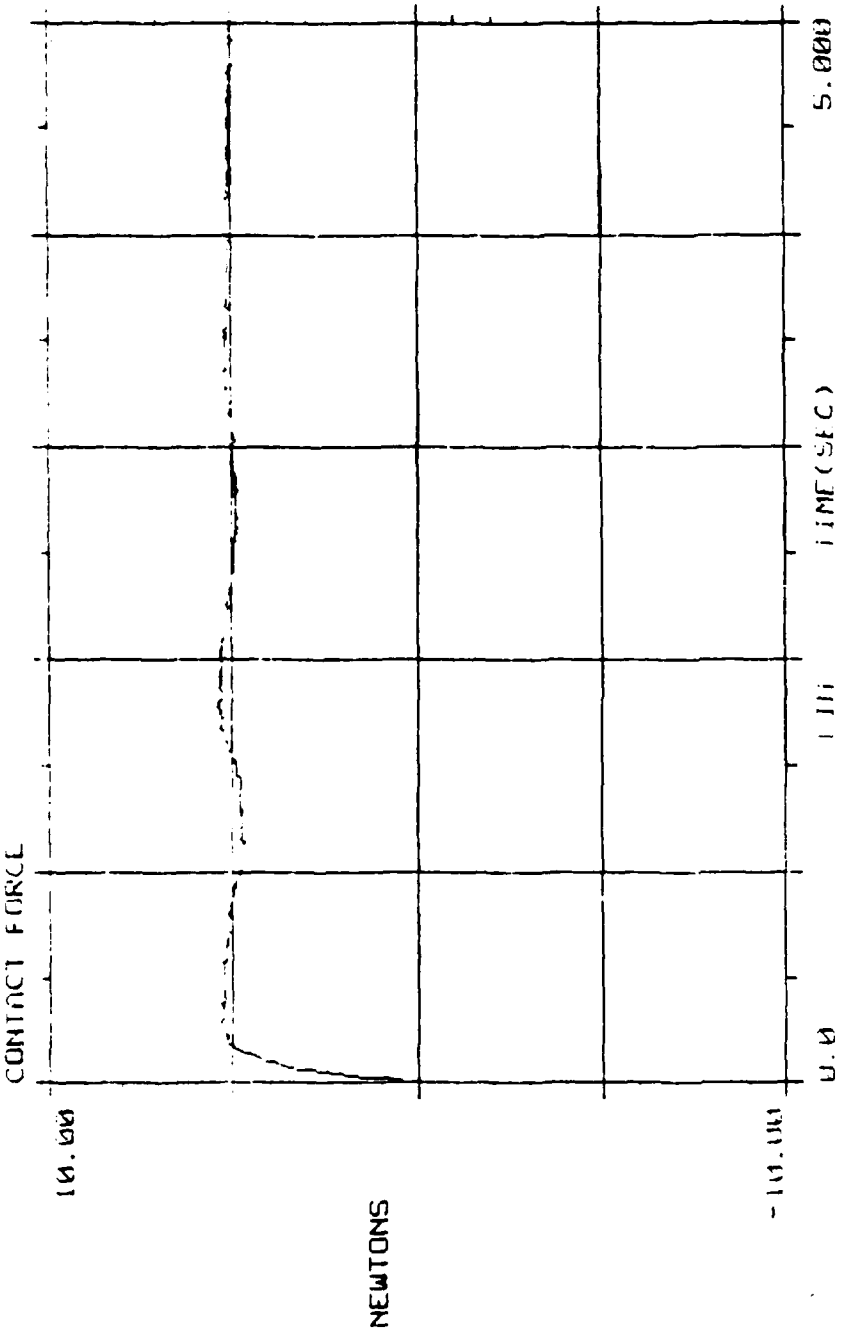
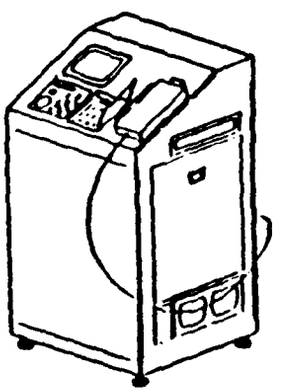
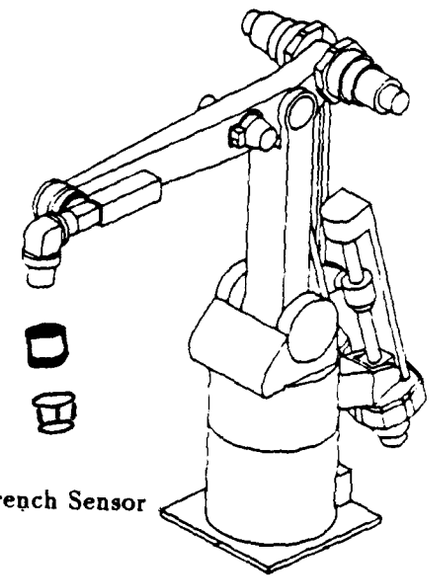


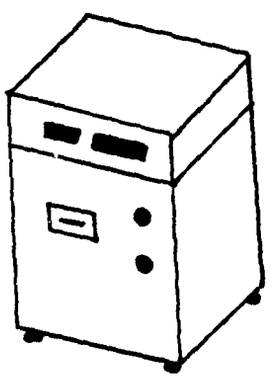
Figure 5.11



RC 1560 Robot Controller

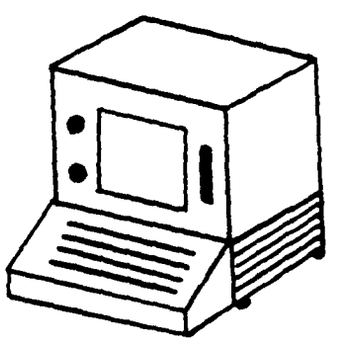


GE P60 Industrial Robot



F/T 15/50

PDP 11/23 Minicomputer



Terminal

Figure 5.12 Integrated System

pendant, the PDP-11/23 minicomputer with auxilliary circuits, as well as the model 15/50, force/torque sensor control panel.

5.5 Results and Conclusions

This chapter covered the system development methodology. Except for the summing circuits required to integrate the force control signals with the velocity commands generated in the RC 1560, all of the hardware used for this work was commercially available in a ready-to-use, final form. Furthermore, even these auxilliary circuits consisted of components readily available at any corner electronics store.

Of particular interest was the system integration and testing. As already observed observed by An, Atkenson and Hollerbach, "Experimentation also stimulates discovery," where new issues evolve "serendipitously from problems with actual implementation" [An *et al.*, 1988, page 2.]. The limit cycle effects observed certainly fell into this category.

With an integrated, tested system, it remained to ascertain performance capability, as well as to actually implement the example tasks formulated in Chapter 4.

CHAPTER 6
EXPERIMENTAL RESULTS

6.1 Introduction and Objective

The principal objective of this chapter is to validate the conceptual design presented in Chapter 4, through the use of the hardware and software systems developed in Chapter 5. This process begins with a presentation of wrench control performance results: step response characteristics, command following bandwidth and disturbance rejection bandwidth.

The chapter continues with an analysis of experimental uncertainty. The author considers that such an analysis is essential in order to ascertain the potential significance of the results.

Finally, the representative industrial tasks that were formulated in Chapter 4 are discussed, along with an analysis of their laboratory implementation.

6.2 Performance Results

It is important to reaffirm the effect of reciprocal wrench constraints on the actuator. Figure 6.1 shows the motor velocity of the vertical axis while the end-effector followed a straight line along a flat surface, both with (Figure 6.1a) and without (Figure 6.1b) a reciprocal wrench command applied. One can readily see that the motor velocity was essentialy unaffected by the application of a reciprocal wrench command. This clearly demonstrated the decoupling of motor torque and velocity, resulting from the reciprocal task constraints. This decoupling is both a fundamental premise and an

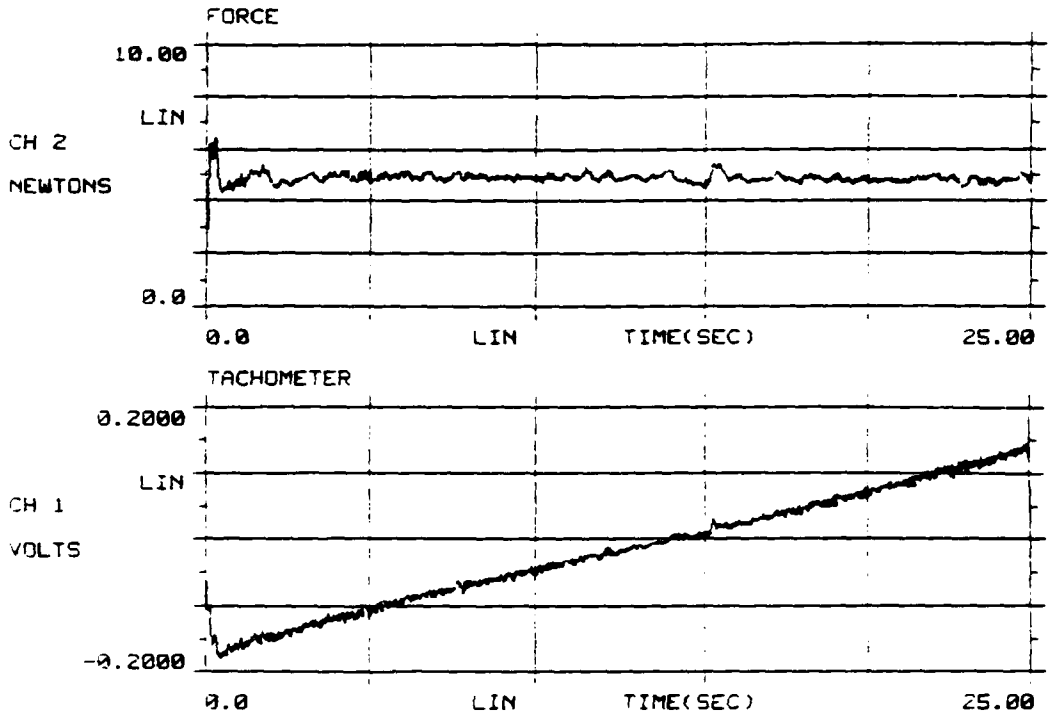


Figure 6.1a

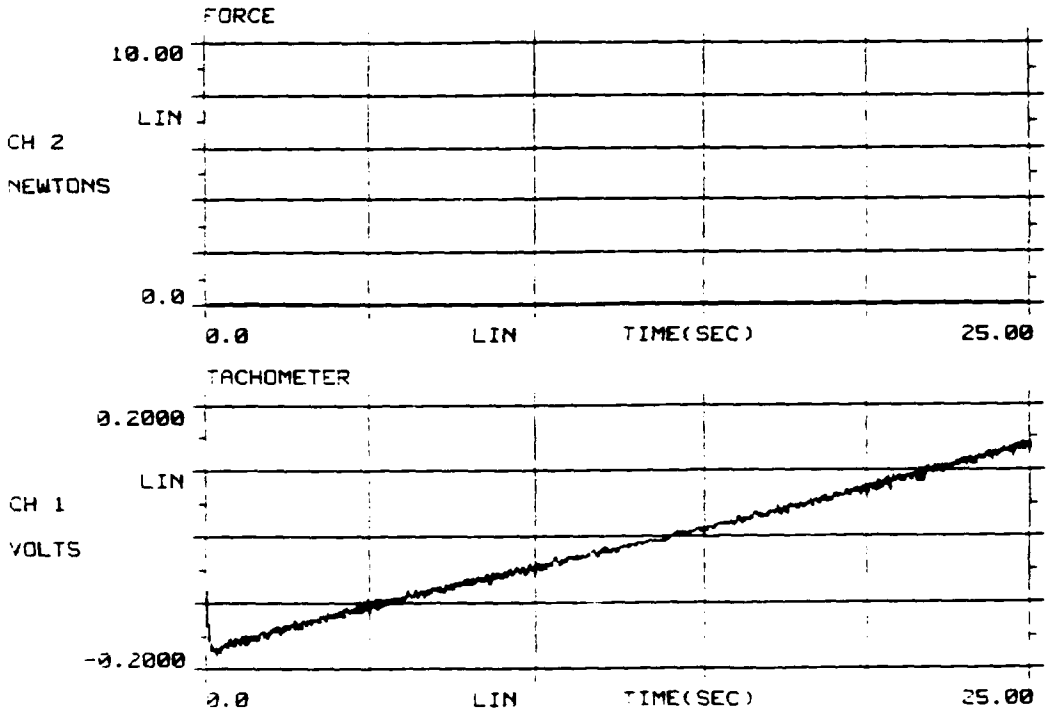


Figure 6.1b

Figure 6.1 Vertical axis response with a) both force and speed control: b) speed control only.

essential feature of cross-coordinated control. Since the motor velocity is unaffected, it follows from Equation 3.1 that the resulting twist at the end-effector is similiarly unaffected. One should recall from Chapter 3 that voltage-controlled robots, such as the GE P60, are speed controlled by design. Hence, it was unnecessary to augment the speed control intrinsic with the commercial system. The simple fact is that convenient sensors to directly measure the end-effector twists are not available. Cross-coordinated control, as proposed in Chapter 4 and implemented here, avoids the necessity for the direct measurement of these end-effector twists, and requires solely the measurement of wrenches acting on the end-effector. Such measurements were easily made with a high degree of accuracy using the wrist mounted, commercial force/torque sensor.

The use of step response parameters is sometimes criticized in the literature [An et al., 1988, and Stepien et al., 1987] for allegedly not being the best indicator of force control performance. Such criticism may be due, at least in part, to the typically poor step response performance usually reported. This is particularly true with respect to overshoot, as discussed earlier.

With the end-effector of the GE P60 in contact with a solid block of aluminum, as shown in Figure 4.14, a step input of 5 Newtons was commanded. The response is shown in Figure 6.2. The rise time is less than 200 msec, while exhibiting negligible overshoot. These are precisely the characteristics required for assembly tasks, such as the peg-in-the-hole problem. While a rapid servo response, as indicated by a short rise time, is desirable for any servo control application, sufficient damping, as indicated by low overshoot, is critical to assembly tasks.

Significant overshoot can be catastrophic for these tasks, such as the peg-in-the-hole problem, where the task geometry is characterized by bilateral constraints.

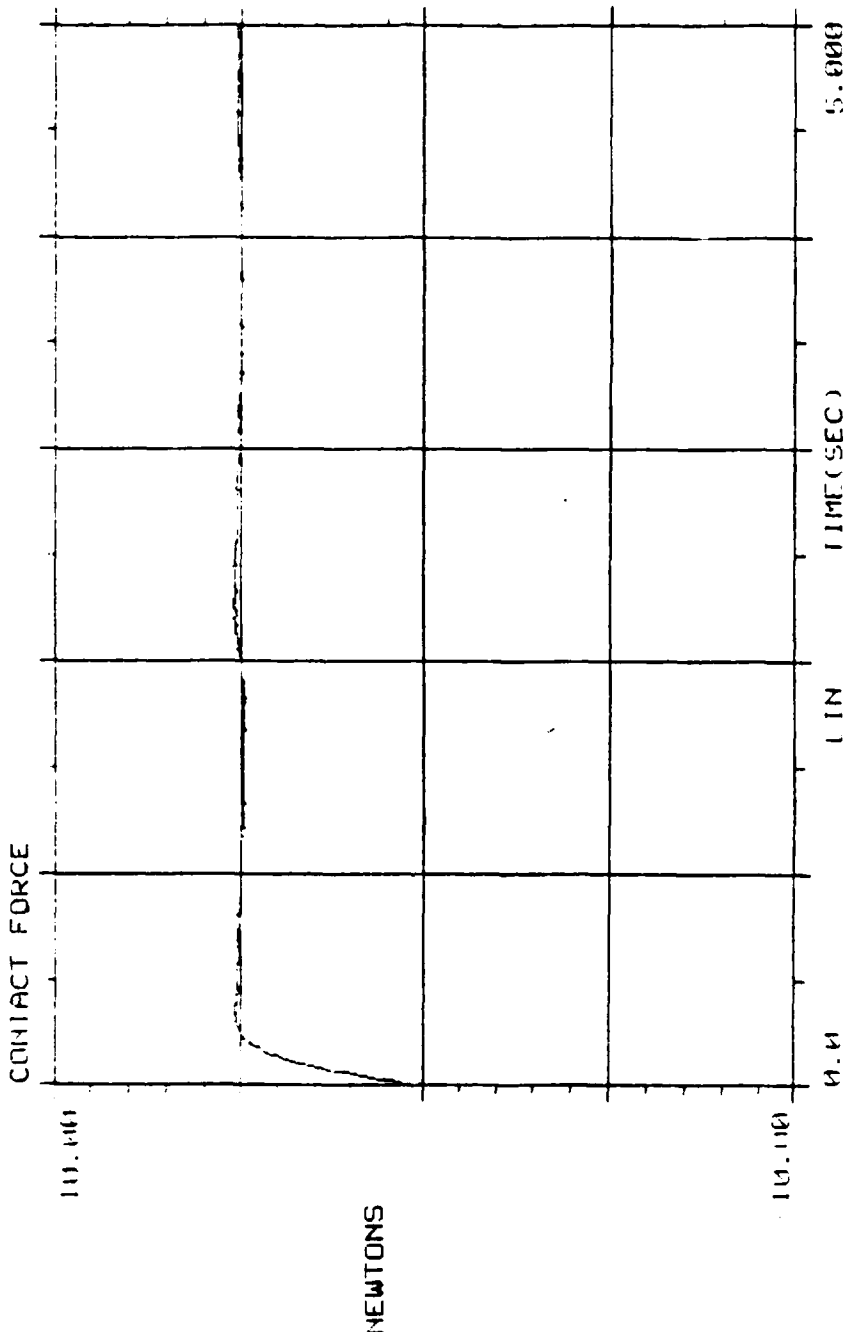


Figure 6.2

Overshoot is also undesirable for process tasks, represented here by contour following, though it is potentially less catastrophic due to the typically unilateral geometry of the task constraints. Nonetheless, these time domain performance parameters are not the only way to evaluate control system performance.

An et al. [1988] recommended two frequency response parameters as being better measures of performance. These parameters are the bandwidths for command following and disturbance rejection.

Command following bandwidth is a measure of the frequency at which the system can no longer adequately follow a command value. One way to determine this value is to generate a Bode plot, using Fast Fourier Transform (FFT) techniques. Using this impulse response algorithm, as implemented on a Gen Rad frequency response analyzer, the Bode plot is generated for the external force control loop on the vertical axis, as shown in Figure 6.3. Based on the usual definition of bandwidth (frequency at which the amplitude reduction from dc value equals 3 dB), this value is about 1.5 Hz. However, this technique is more difficult to implement for a multi-axis strategy, such as cross-coordinated control, since the FFT algorithm is primarily intended for a SISO system.

An et al. [1988] suggested the more traditional (and flexible) technique of directly applying a sine wave command of varying frequency to the system, and then measuring the output waveform to determine the frequency at which the response to command signal had sufficiently degraded. One may recall that this was the approach used to generate Bode plots before frequency response analyzers became so readily available. This direct, if somewhat more time consuming, approach can be applied more easily than FFT techniques to multi-axis implementations, since the input signal is finite

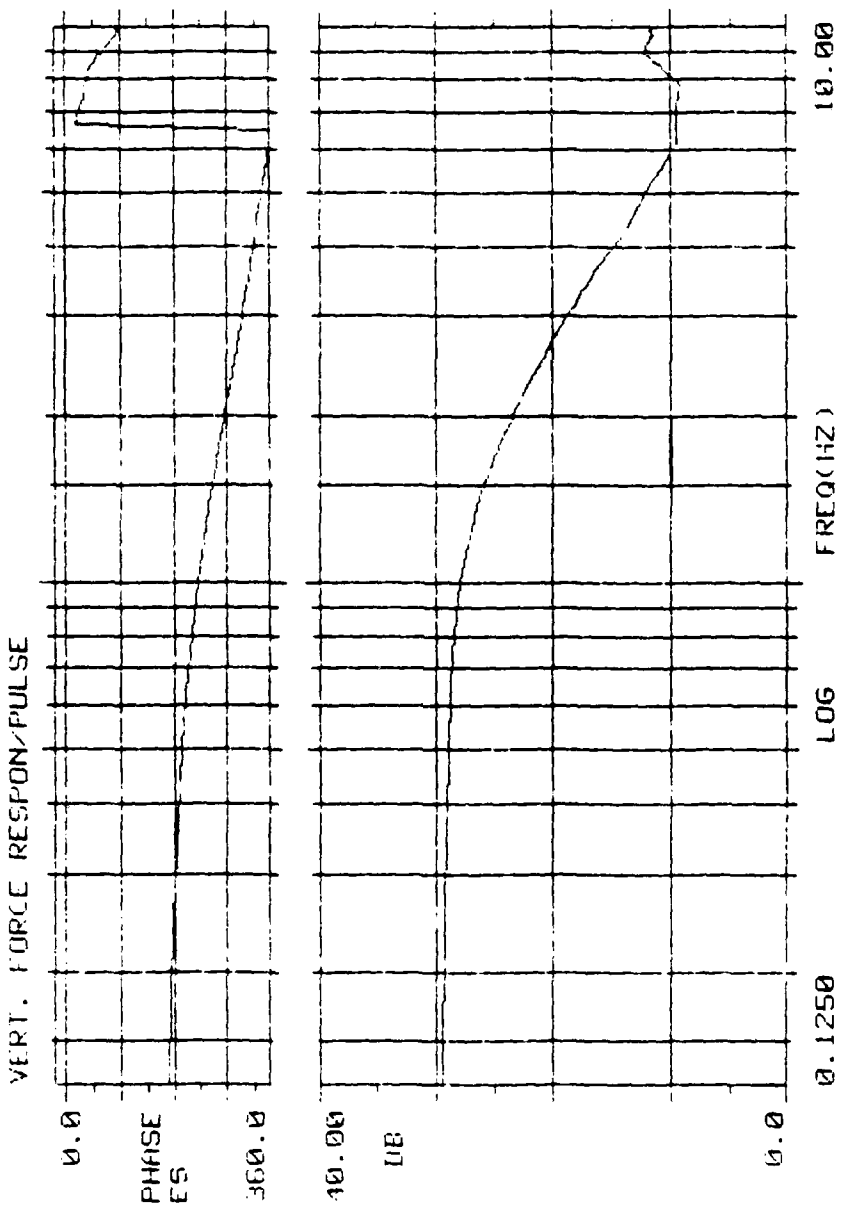

VERT. FORCE RESPON/PULSE
0.0
PHASE
DEGREES
360.0
40.00
DB
0.0
0.1250
LOG
FREQ(HZ)
10.00

Figure 6.3

by design, rather than a finite approximation of an infinite pulse of infinitesimal duration.

The author has strong reservations, however, about the test procedure used by An et al. [1988] to evaluate this parameter on their direct-drive robot. These stem from the fact that they tested for this value with the robot *essentially stationary*. One could reasonably expect this approach to overestimate the useful command following bandwidth for two reasons.

Firstly, a highly underdamped system, such as that used by An et al [1988], will tend to amplify the command value long before the 3 dB reduction occurs. Had the criteria been a 3 dB variation, it appears, judging by their test results, that the bandwidth would have been closer to 5 Hz than the 20 Hz they reported, and even this estimate is probably a little optimistic.

Secondly it is important to note that when they tested their system in a contour following mode, viz. employing a force command along one Cartesian axis and a velocity command along the mutually perpendicular axis, the bandwidth dropped still further, this time to about to 1 Hz. Since this latter set of test conditions, for which the robot was in motion while responding to a force command, included the effects of link inertia, they will certainly provide a much more conservative estimate of performance. The reader should also note that An et al. [1988] never attempted to apply force control to more than one actuator.

For this work, a sine wave force command was applied to the GE P60 system, with the results shown in Figure 6.4. The bandwidth was found to be about 1 Hz. Interestingly, this value was essentially the same regardless of whether the end-effector was stationary or traversing the surface of the aluminum block. The author believes

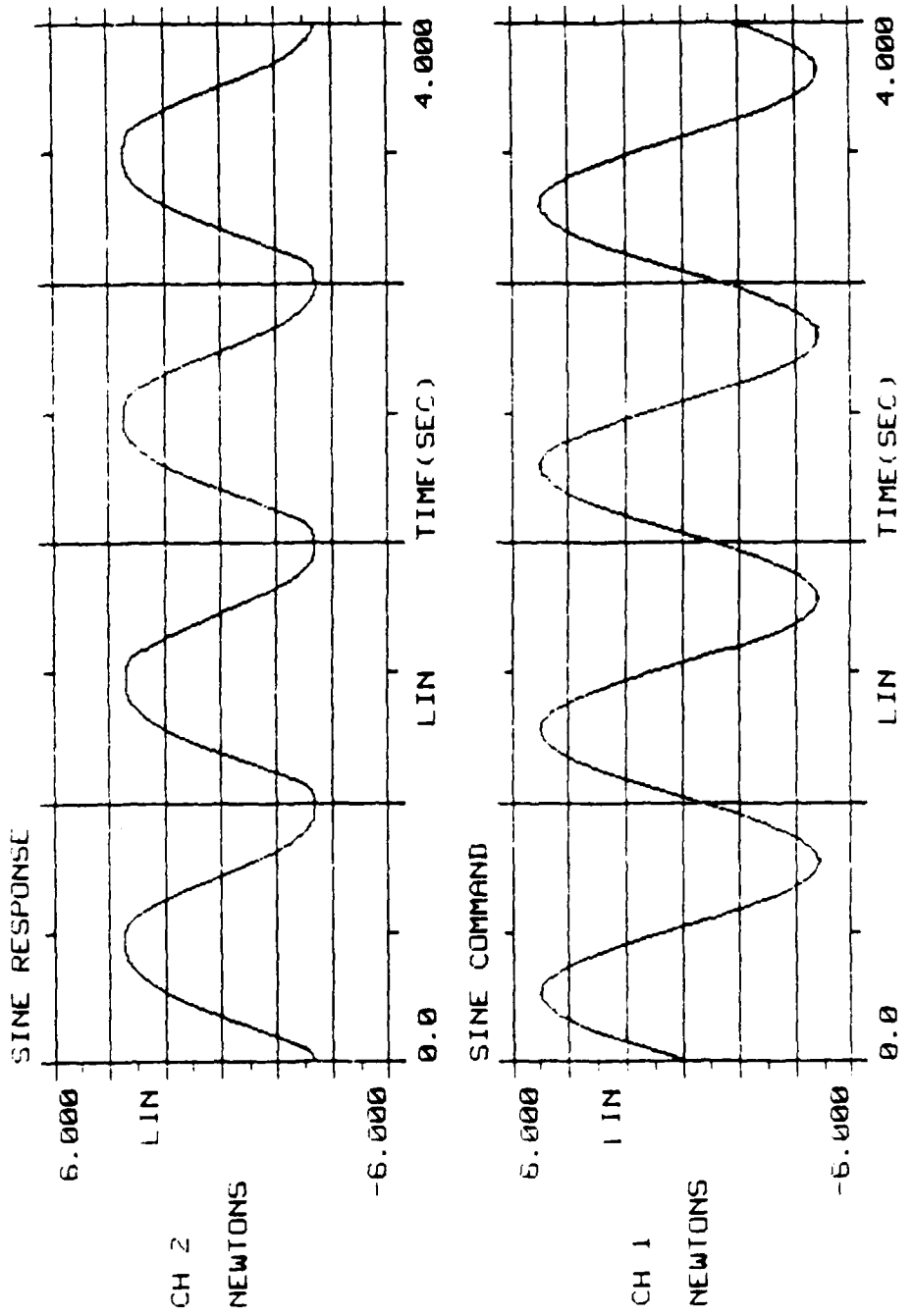

SINE RESPONSE
6.000
LIN
CH 2
NEWTONS
-6.000
0.0
LIN
TIME(SEC)
4.000
SINE COMMAND
6.000
LIN
CH 1
NEWTONS
-6.000
0.0
LIN
TIME(SEC)
4.000

Figure 6.4

this to be due primarily to the highly geared transmissions, which cause the link inertias to be dominated by rotor inertias. The use of highly damped, redundant actuators may also play a role in moderating inertia effects.

Disturbance rejection was the other performance test suggested by An et al. [1988]. The author has similiar reservations concerning their test procedure used to evaluate this parameter. They essentially applied a constant force command to the end-effector while it was in contact with a motor driven, eccentric cam. By turning the cam, an approximate sine wave disturbance could be generated. Varying the speed of the cam rotation adjusts the frequency of the disturbance. While this test rig is ingeneous, one would suspect that it will also tend to significantly overestimate the disturbance rejection bandwidth since, once again, the manipulator is essentially stationary.

Further, this testing procedure has the disadvantage in that tends to eliminate nearly all of the link inertia effects which would normally be present when all the robot links are in motion. One should recall that direct-drive robots, like the one used by An et al. [1988], lack the highly geared transmissions found on industrial robots. Hence, the varying link inertias are not dominated by the essentially constant rotor inertias in the direct-drive design. Therefore by testing an essentially stationary robot, a significant disadvantage of the direct-drive design is largely masked by the testing strategy, namely the inherently strong sensitivity to varying link inertias.

An et al. [1988] suggested an alternative bandwidth criteria for disturbance rejection, namely the frequency where

$$\frac{f_{command} - f_{response}}{f_{command}} = 10\% \qquad (6.1)$$

This criteria does represent stricter bounds than the usual definition. However, by again considering only reduction in amplitude of the response, rather than variance from the command value, this criteria favors an underdamped servo. Using this criteria, they reported a bandwidth of 0.8 Hz for disturbance rejection. Again, by having their characteristically heavy, direct-drive manipulator essentially stationary for this test, and thus eliminating most of the link inertia effects that would be seen with their robot in motion, their results would have almost certainly overstated the usable bandwidth for most applications. However, it must be acknowledged that unlike 99% of the research published on robot control, An et al. [1988] actually performed extensive testing, and fully documented both their results and the test procedures used.

The approach taken by the author to test disturbance rejection was somewhat different. Rather than having the manipulator stationary, and apply a disturbance with a cam, the configuration shown in Figure 6.5 was used. A sine wave of amplitude $\pm$ 0.1 inch was milled into the edge of the aluminum block. The robot was taught a start point and a stop point along this undulating surface, and commanded to traverse a straight line between the two taught points, while simultaneously applying a constant force of 5 Newtons which was reciprocal to the commanded twist. By varying the translational speed of the roller, the frequency of the sine wave disturbance could be modified. It is important to recognize that this test configuration, which was similar to that used by the author for command following, included the full effects of robot link inertias. Using this approach, the disturbance rejection bandwidth (based on the usual definition) was observed to be about 0.5 Hz as shown in Figure 6.6.

The author strongly suspects that had he used a test configuration for disturbance rejection similiar to that used by An et al. [1988], the results would have

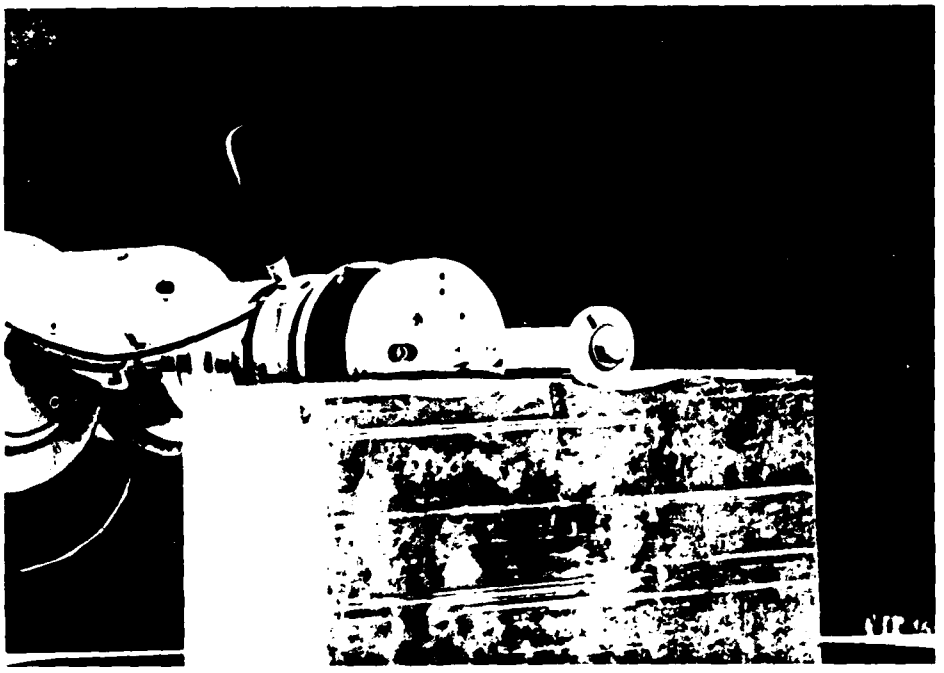


Figure 6.5

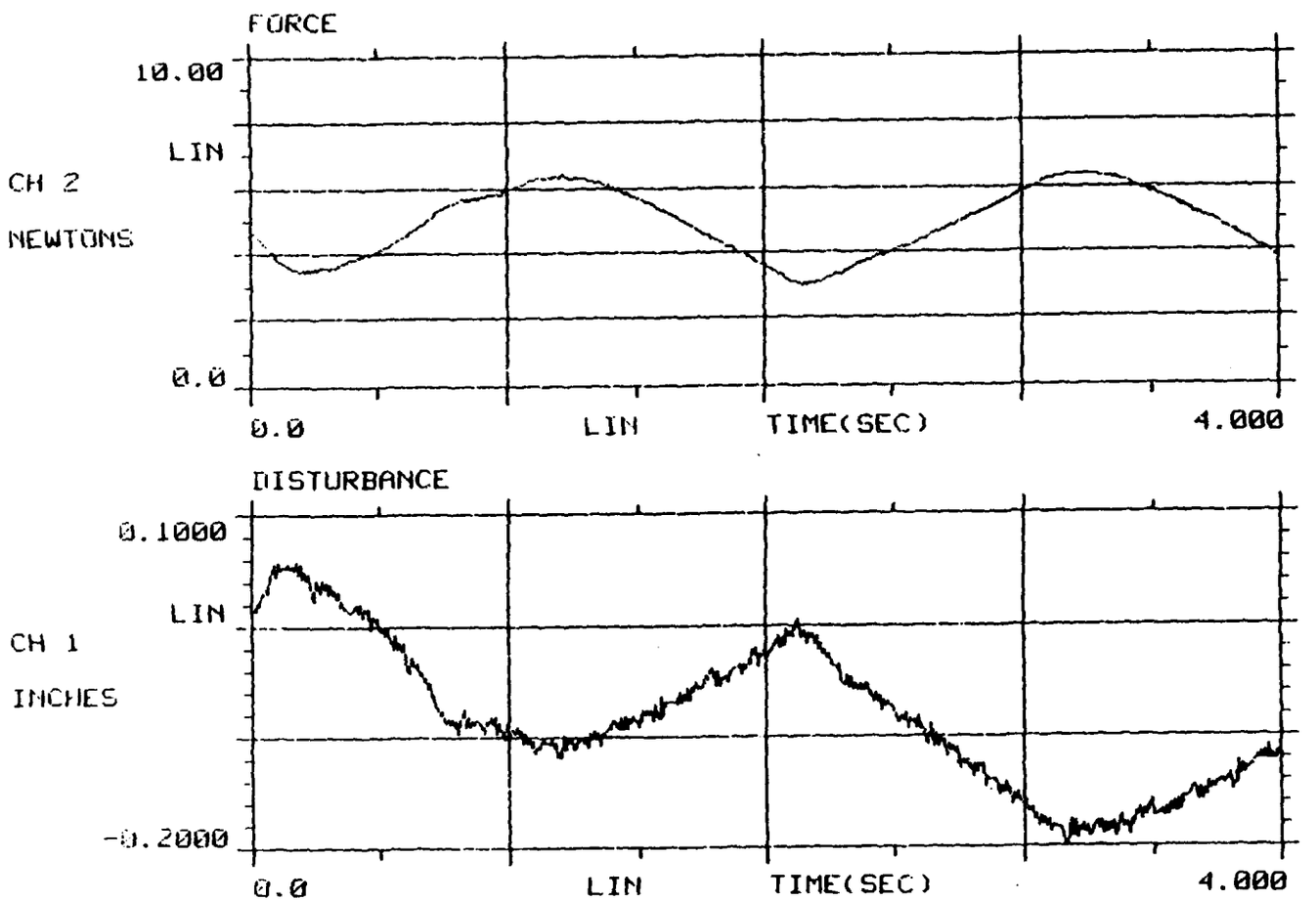

FORCE
10.00
LIN
CH 2
NEWTONS
0.0
0.0
LIN
TIME(SEC)
4.000
DISTURBANCE
0.1000
LIN
CH 1
INCHES
-0.2000
0.0
LIN
TIME(SEC)
4.000

Figure 6.6

been substantially the same as those observed with the test configuration just described. The reason is that modern industrial robots employ characteristically high transmission ratios and velocity feedback. Consequently, industrial robots (unlike direct-drive robot designs) are extremely insensitive to the effects of link inertias. Since it is largely the capability to perform reprogrammable motion which justifies the higher expense of robots over conventional fixed automation, it would seem that performance parameters, based on an essentially stationary robot, are inappropriate.

6.3 Uncertainty Analysis

The term experimental uncertainty is introduced here, to distinguish it from experimental error, although they are often interchanged in the literature. The author firmly believes that this analysis should always be performed, whether it takes the form of complex statistical analysis, or merely a verbal assessment of the results by an experimenter. Some assessment of experimental results is needed, since it is only the experimenter who will have a first-hand sense of the validity of his data.

Some errors will always creep into experimental work. Usually these errors are random, although an occasional major error may occur. Fortunately, such errors are usually easy to detect. Standardized criteria for eliminating those data points, such as Chauvenet's criterion, are well known [Holman, 1984]. However, random error, referred to here as uncertainty, is always present to some degree. The goal is to determine that degree, which is usually evaluated through standard statistical techniques.

The reason for the introduction of the term uncertainty to describe random error is one of clarity. An experimental error is, after all, still an error. If the experimenter recognizes an error, he will eliminate it. The real errors which plague good experimental work are those that are due to uncertainty, and are inherent in any laboratory work.

In Chapter 3, the author used the Student's t statistic to evaluate the confidence interval for the plant model used to design the compensator, and a 90% confidence interval was selected. The choice is of course subjective. However, it can be used as a measure of the relative validity of the model.

Other techniques to evaluate experimental data are available, especially for multi-sampled data (data obtained by repeating experiments). The processing of such data using standard statistical techniques is relatively simple. As an example, consider that the rise time for the step response to a force command of 5 Newtons was reported to be less that 200 msec. In fact, the mean value was 197 msec, based on one hundred replications of the test. The variance was calculated to be 2.4 msec. As further examples, consider that the command following bandwidth was found to be 0.98 Hz, with a variance of 0.05 Hz, while the disturbance rejection bandwidth was found to be 0.52 Hz, with a variance of 0.08 Hz. Both of these examples were also based on one hundred test replications, with the robot manipulator in various configurations.

Intermediate results, such as the determination of gear ratios, were similiarly analyzed. Without such analysis of intermediate results, the experimenter will surely find it difficult to track down the cause of any unexpected variations in the final results. In fact, this type of analysis is essential to the process of system integration and testing, as discussed in Chapter 5.

It is a more difficult problem to analyze single-sample data. Nonetheless, cost will sometimes prohibit multiple experiments, as will a tight schedule, or the need for destructive testing. Since single-sample testing was not required for this work, these techniques will not be discussed. However, interested readers should refer to Kline and McClintock [1953] for a description of some of the techniques available.

The two major sources of uncertainty for these experiments appeared to be the resolution of the D/A board and the resolution of the force/torque sensor. The D/A board had 12 bits of resolution, with a range of ± 10 volts. This means that this span of 20 volts could be resolved into 4096 increments. As already mentioned, the servo amplifier circuit would only permit an augmentation signal of $\pm$ 1 volt. At first, it may seem that this would lower the resolution to about one part in 410. However, this was not the case. The control signal was in fact generated using the full 12 bits of resolution over the full, 20 volt output range. However, before this signal was added to the circuit, it was passed through an analog voltage divider with a factor of 1/10. Since the voltage divider was an analog circuit composed of high precision resistors, virtually no resolution was lost.

The resolution of the force/torque sensor is indicated in Table 6.1. With the end-effector in contact with the aluminum block such that the contact force was tangential to the arc traced by an incremental motion of the bend axis, a change of 100 D/A counts (100 bit change in the output signal) produced a change of about 35 uf (units of force) in f_y for the open-loop system, where f_y and f_z are respectively the forces in the directions normal and parallel to the workpiece surface. These force units were quinta-ounces (1/5 ounce). Note that f_z was never used for active force control, since it lies along the stiff axis of the RCC device. This input/output relation for the bend axis means that the D/A board has about 2.5 times the resolution of the force transducer for this configuration. This ratio was similiar to the ratio for the roll axis. A similiar result was obtained for torque about the z-axis, which was dominated by the twist axis response.

Table 6.1

Lord Corporation Force/Torque Sensor Model 15/50		
Component	Resolution	Uncertainty
f_x	1 uf*	$\pm$ 0.50 uf
f_y	1 uf	$\pm$ 0.50 uf
f_z	3 uf	$\pm$ 1.50 uf
t_x, t_y, t_z	2 in-uf	$\pm$ 1.00 in-uf

* uf: units of force are 1/5 ounce

The situation for the large motors, namely the resolution ratios for the vertical, horizontal and swing axes, was essentially reversed. Here, configured as just described, the force sensor had 2-4 times the resolution of an incremental change in voltage applied to these large motors. In either case, however, the problem was not significantly detrimental to seriously degrade performance. This is demonstrated by the following analysis.

The uncertainty of the steady state, closed-loop values of the wrench components are simply those of the force/torque sensor, as listed in Table 6.1. The uncertainty associated with the D/A outputs is not an issue under these conditions. However, during the transient response, this is not necessarily the case. Between samples, digital servos may be modelled as open-loop systems.

If one assumes that under these conditions, the output force is a function of the D/A outputs of the form,

$$R = R(x_1, x_2, x_3, \ldots, x_n) \tag{6.2}$$

and the uncertainties of each of these independent variables, w_i, is known to the same level of confidence, then the uncertainty of result, w_r, can be estimated [Kline and McClintock, 1953] using

$$w_r = \left\{ \left(\frac{\partial R}{\partial x_1} w_1 \right)^2 + \left(\frac{\partial R}{\partial x_2} w_2 \right)^2 + \cdots + \left(\frac{\partial R}{\partial x_n} w_n \right)^2 \right\}^{1/2} \tag{6.3}$$

Furthermore, assume that the steady state, open-loop force response can be described by

$$R = 0.72\, x_1 + 2.60\, x_2 + 1.84\, x_3 \tag{6.4}$$

where x_1 is the D/A input to the bend axis, x_2 is the D/A output to the vertical axis, and x_3 is the D/A output to the horizontal axis. Using Equation 6.3 and Equation 6.4, and assuming that a signal of 100 D/A counts was sent to each of the bend, vertical and horizontal axes, the resulting steady state force would be 516 uf with an associated uncertainty of 1.63 uf.

6.4 Examples of Industrial Applications

As already noted, nearly all industrial robots are currently employed in low precision tasks, such as pick-and-place, spray painting or spot-welding. While there has been much focus on sophisticated control algorithms intended to permit increased speeds

without significant degradation in repeatability, research at General Electric, as reported by Sweet and Good [1984], indicates that even a 100% increase in peak robot speeds (unlikely because of limits on motor torque) would likely have only a marginal effect on the economics of a robot installation decision. On the other hand, robots that can compensate for some level of positional uncertainty in manufacturing applications have an enormous potential for expanding the spectrum of tasks for which robotic installation is economically justified. Two of those areas where this accomodation capability is essential are assembly tasks and contact process tasks.

Assembly tasks generally require the solution of fairly high precision positioning problems. Usually, some non-zero constraint wrench serves to maintain contact between parts, while zero valued wrench constraints are used to prevent jamming. Shahinpoor [1987] reports that fully 35% of all assembly applications involve some kind of insertion of a round peg into a hole. It was for this reason that the author chose this type of problem to demonstrate the utility of cross-coordinated control for assembly tasks.

As discussed in Chapter 4, the insertion problem really has two phases, tip insertion and shaft insertion. Two alternative methods for tip insertion were demonstrated in this research. These were chamfer assisted tip insertion and the development of a tipping strategy for unchamfered tip insertion. When the chamfer width, as shown in Figure 6.7, exceeds the total positional uncertainty of the task, then it, together with the engineered mechanical compliance provided by the RCC, will usually be enough to ensure tip insertion. This engineered compliance will dominate any undocumented compliance that is systemic to a well designed industrial robot. The unchamfered hole problem can be handled using the tipping strategy discussed in Chapter 4. The tilted peg approaches the lip of the hole until contact, as shown in Figure 6.8. The peg then

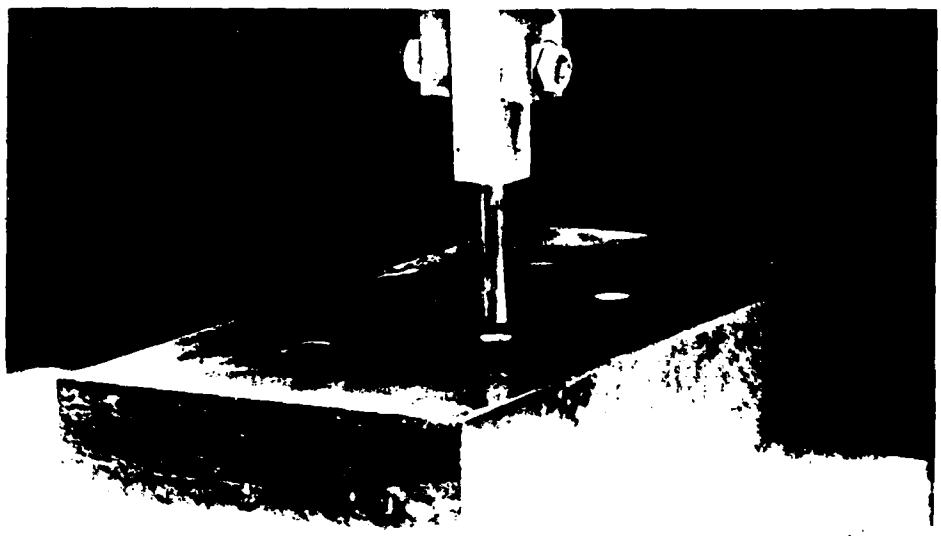


Figure 6.7

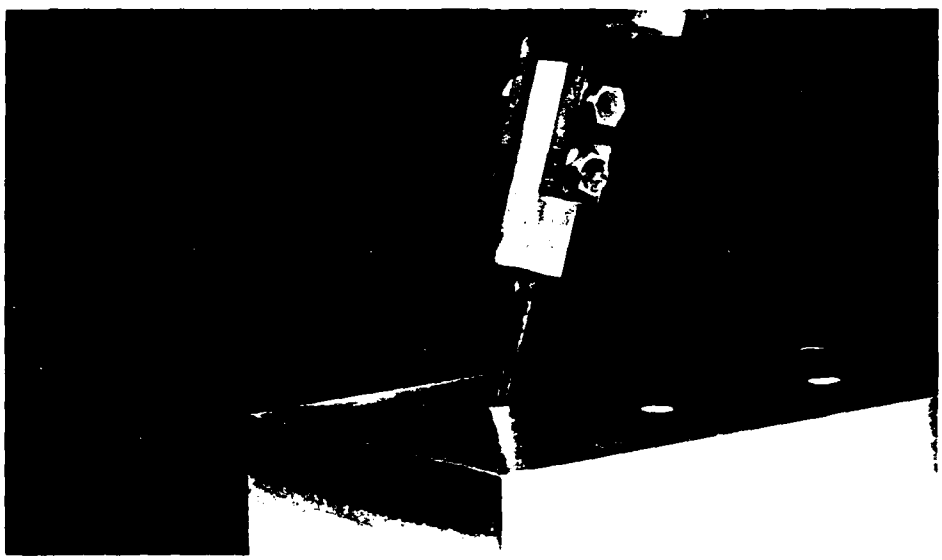


Figure 6.8

applies a reciprocal, non-zero wrench to the lip of the hole in order to maintain this one point contact, while the peg rotates about its tip until its axis is aligned with that of the hole, as shown in Figure 6.9a. Once the tip is in the hole, a zero-valued constraint wrench, as developed in Chapter 4, precludes jamming as the peg is inserted. (See Figure 6.9b)

Contact process tasks are those applications for which the tool must follow a trajectory, while applying a specific wrench to the workpiece. There are two factors which distinguish this type of problem from assembly operations. First, the task geometry typically provides only a unilateral constraint, as opposed to the fully bilateral constraint of a peg-in-the-hole problem. Second, the magnitude of the wrench, itself, is usually a design parameter.

Grinding, for example, involves the fairly complex dynamics of an abrasive or cutting tool to remove material from a workpiece. The metal removal rate depends on several parameters. Once the tool rotation speed is selected, however, the kinestatic constraints are governed by the normal force and the relative translational velocity of the tool over the workpiece, as depicted by Figure 6.10. Tlusty [1972] defines a metal removal rate parameter, λ_w, in terms of the metal removal rate, Z, and the average normal force F_n as

$$\lambda_w = \frac{Z}{F_n} \tag{6.5}$$

For a given grinding speed and tool speed, this normal force is related to a stiffness parameter, K, and the depth of cut, d, by

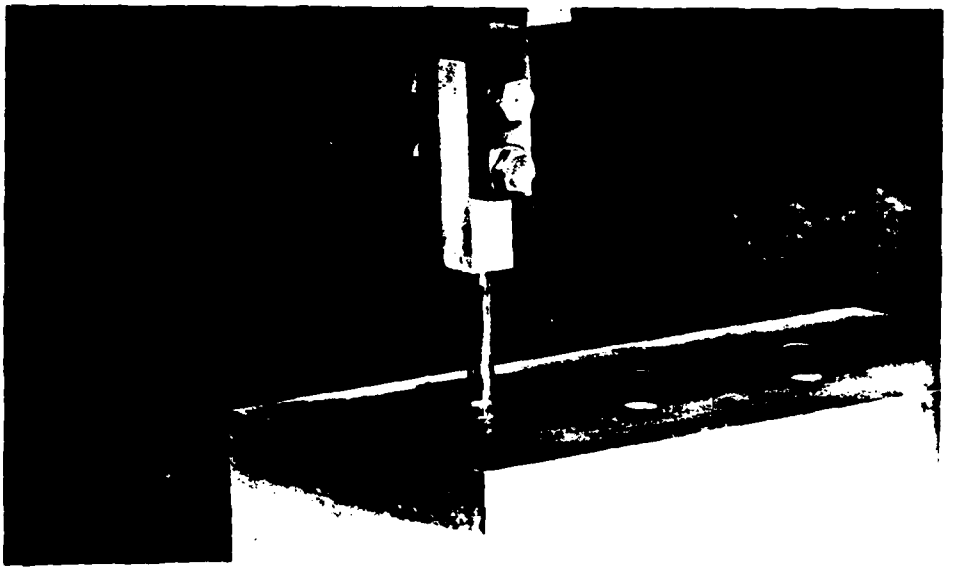


Figure 6.9a

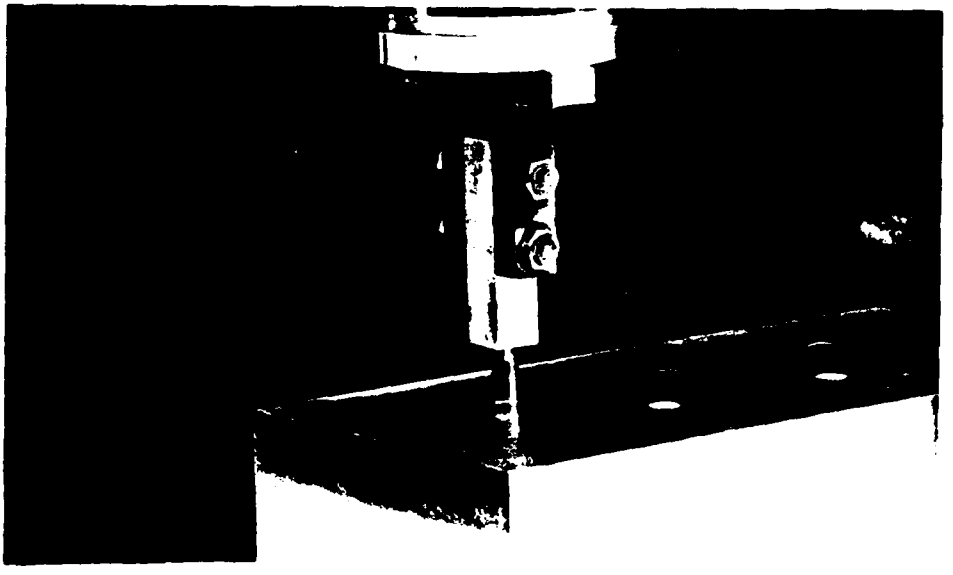


Figure 6.9b

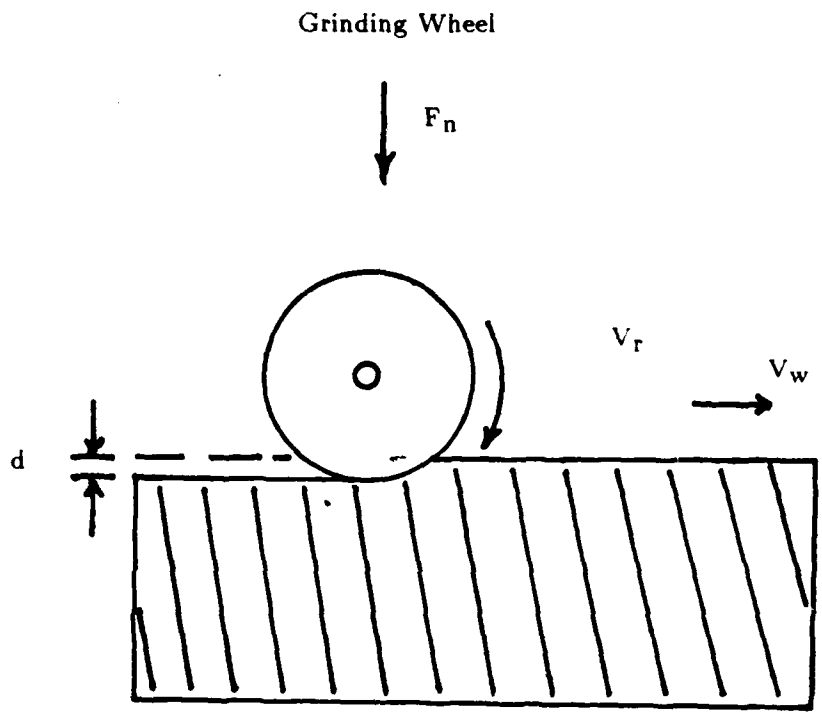


Figure 6.10

$$F_n = K \cdot d \tag{6.6}$$

while the metal removal rate, Z, is related to the translational velocity of the tool over the workpiece, V_w, and width of cut, B, by the relation

$$Z = d \cdot B \cdot V_w \tag{6.7}$$

It is important to recognize that the rotational speed of the grinding wheel, on the order of several thousand times the bandwidth of the force controller, is so fast as to have very little dynamic effect on the control loop. For this reason, this task was modelled as a contour tracking problem.

Similar to the grinding or deburring problem, contour tracking requires the control of the tool speed as it transverses the workpiece, as well as the application of a normal force, as shown in Figure 6.11. Unlike grinding or deburring, however, the value of this normal force is usually not a design parameter. It is simply required to maintain contact. One should note that contour tracking is commonly used in applications which require the use of templates. When the tool's orientation is also critical, an alternative, double-wheeled roller fixture, as shown in Figure 6.12a, can be used.

By specifying a non-zero torque about the central axis of this end-effector, as discussed in Chapter 4, the end effector will maintain a constant angle of attack to the contour surface, as shown in Figure 6.12b. A tool, rigidly attached to this double roller, will also maintain this orientation.

Although these examples are not exhaustive, they do cover a wide range of industrial tasks. The results obtained strongly indicated that there is a wide range of

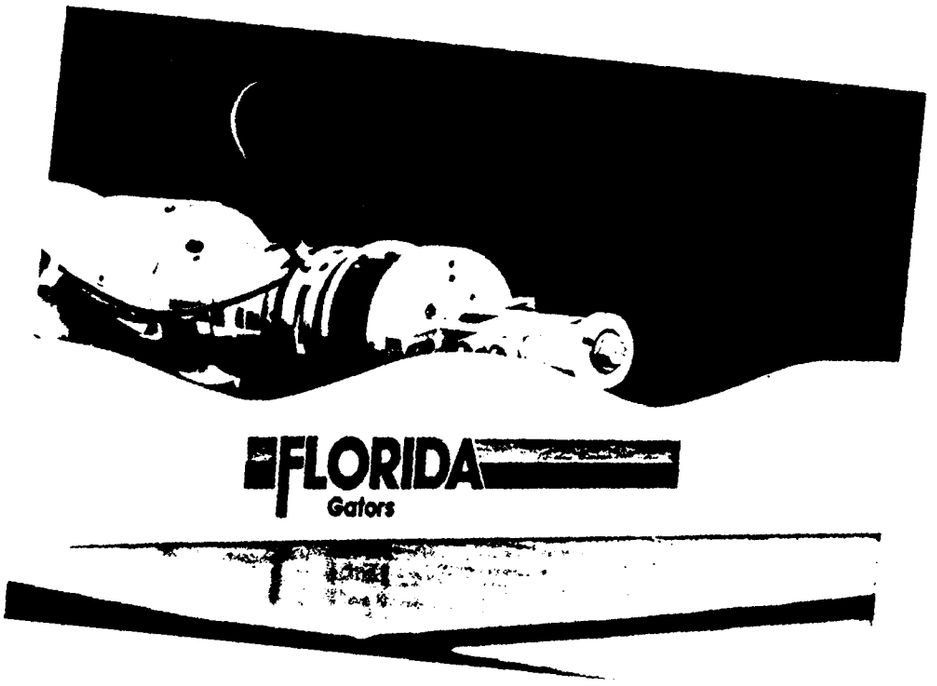


Figure 6.11

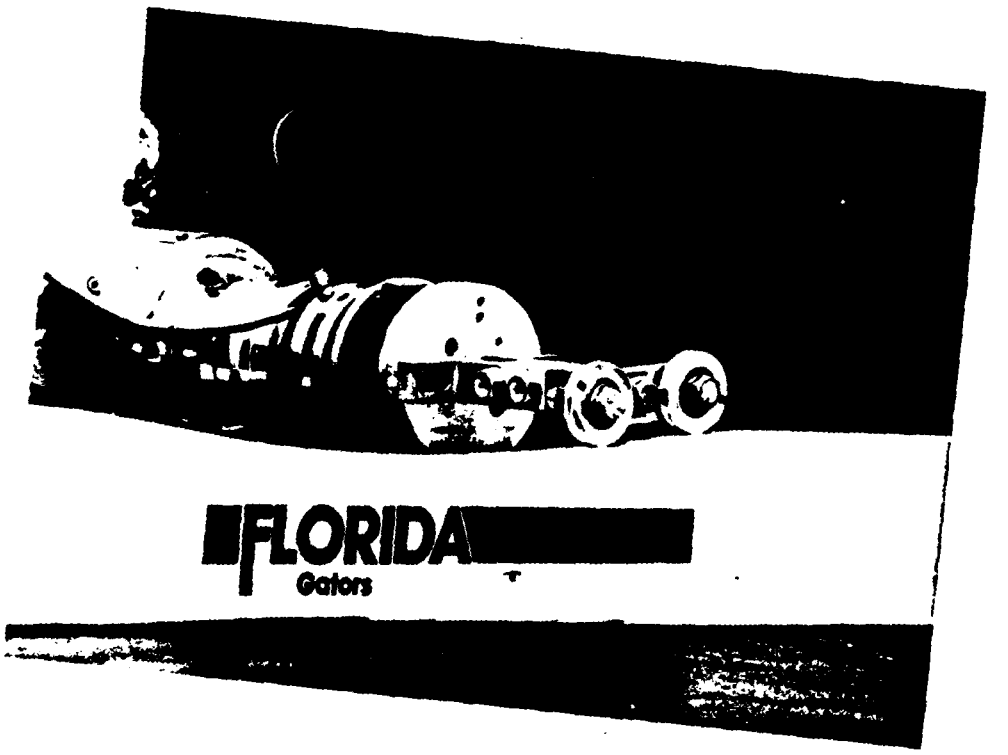


Figure 6.12a

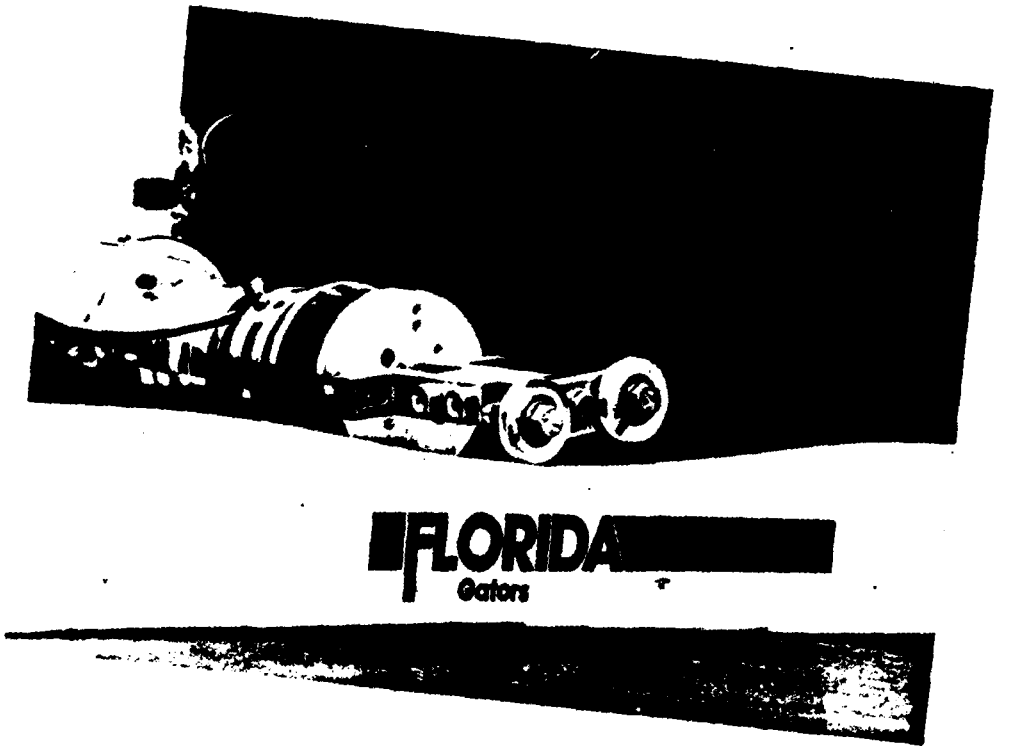


Figure 6.12b

applications for voltage-controlled industrial robots which, heretofore, were generally considered to be outside their capabilities [Snyder, 1985, and Koren, 1985]. It appears that such additional capabilites will translate into an expanded range of economically justified, industrial robot installations.

6.5 Results and Conclusions

This chapter presented specific performance results. These consisted of both time domain parameters, as characterized by step response data, as well as frequency response characteristics, namely the bandwidths for command following and disturbance rejection. The author was careful to describe in some detail the test procedures used to obtain these results. This provides a basis for comparison with other experimental results.

An uncertainty analysis was performed. This analysis described the methods used to quantify experimental uncertainty for this research, thus aiding the reader in evaluating the potential significance of the results.

Finally, the representative industrial tasks selected for the experiments, the models for which were formulated in Chapter 4, were successfully implemented in the laboratory. These tangibly constrained tasks represent a whole new range of potential applications for voltage-controlled industrial robots, such as the GE P60.

CHAPTER 7
DISCUSSION AND CONCLUSIONS

A hybrid twist and wrench control strategy for industrial manipulators, which has been named cross-coordinated control, has been successfully developed and implemented. This strategy, which employs screw theory to formulate geometrically meaningful constraints [Lipkin and Duffy, 1988], has been experimentally verified in the laboratory with a voltage-controlled industrial robot, the GE P60. Performance results, including an analysis of test procedures and experimental uncertainty, have been presented. It is important to recognize that this approach can be readily implemented as an augmentation to an existing robot control system. A major goal of this research has been achieved, viz. the enhancement of existing robot control systems, without the need to replace expensive motion control hardware and software.

There has been a major emphasis in this research on practical issues. This emphasis permeated the system analysis, the conceptual design and the system development. Every effort was made to focus on finding solutions to real problems, rather than what could be described as more pedagogically interesting problems of questionable practical significance. Several representative industrial tasks were implemented in the laboratory.

A further important feature of the system developed here is its general applicability to both assembly and contact process tasks, as well as the commercial availability of the constituent subassemblies. It appears that the model 4 IRCC, which

uses optoelectronic sensors [Seltzer, 1986], has the potential to perform insertion tasks somewhat faster than the device developed here. However, the model 4 IRCC is not only not commercially available, but, its applicability to process tasks will be limited by the relative lack of ruggedness of the optoelectronics compared to strain gauges. On the other hand, for process tasks, there are commercially available devices that may be more effective for grinding and deburring operations than the wrench sensor developed here, such as, for example, the micro-manipulators recently developed by the 3M Corporation [Graf, 1988]. However, those devices are totally unsuitable for assembly tasks. It is the author's considered opinion that it is this ability to perform both process and assembly tasks which makes cross-coordinated control, together with its compliant wrench sensor, such an attractive alternative.

This research is distinguished from recent work by An et al. [1988] and by Asada and Youcef-Toumi [1987], in the sense that it was performed on a commercially available industrial robot, rather than employing a custom-built, direct-drive design. These direct-drive designs may one day become the future industrial standard, but the fact remains, no motor-driven, commercially available, six degree-of-freedom industrial robot currently uses the direct-drive architecture. This is not really surprising, given the current state of the art. Indeed, An et al. [1988] reported problems in that it was not possible to operate their design continuously for a "lengthy period" since a continuous load, like the one caused by gravity, could not be tolerated. Similarly, the few, special purpose manipulators which employ some features of the direct-drive architecture have generally been restricted to motion in the horizontal plane, in order to prevent gravity from acting on the motors. For example, the Adept One robot, manufactured by ADEPT Technologies Inc., and often cited as an example of an industrial, "direct-drive"

robot, relies on steel band transmissions to drive two of its three rotary joints, thus facilitating this horizontal, planar motion. Obviously, such devices are a long way from being practical alternatives to current industrial manipulator designs. Nonetheless, the work completed by these researchers, and especially those two recent books reporting their results, are among the most useful references available today on the state of the art of meaningful robot control research.

It is important to recognize that this research is distinguished from the work done by De Schutter [1986] and De Schutter and Van Brussel [1988], in that it uses sound geometric principles to formulate the constraints, rather than employ intuition based on an erroneous notion of orthogonality. Also, cross-coordinated control was demonstrated here on a modern, voltage-controlled industrial robot, rather than employing an older design, such as the eletro-hydraulic T3-556 employed by De Schutter [1986] and De Schutter and Van Brussel [1988]. Hydraulic robots, like current-controlled robots, are generally considered to be suitable for torque sensitive tasks [Synder, 1987], but unsuitable for speed-sensitive applications. It has been demonstrated here that for tasks for which both speed and torque must be controlled, a speed-controlled robot can be readily augmented with force control. Correspondingly suitable task space speed sensors are not available to augment torque-controlled robots. Nonetheless, the work of De Schutter [1986] and De Schutter and Van Brussel [1988] is extremely important, as it clearly demonstrates the significance of separating the constraint formulation process from the servo control process, as originally conceived of by Mason [1981].

Finally, this work is also distinguished from the work performed by Stepien et al. [1987]. The work here is applicable to a wide range of tasks, both contact process and assembly, and for tasks modelled by both simple and complex geometries. Nonetheless,

their pioneering work was among the first to suggest that voltage-controlled industrial robots, if properly controlled, could be applied to tasks that required contact with a rigid environment.

This research, as well as the work done by those just mentioned, falls into the emerging field of engineering referred to as "mechatronics" [Fourney, 1987]. This multi-disciplinary field is concerned with the electronic control of mechanical systems, with industrial robots being a prime example. Like the emergeance of aerospace engineering as a separate engineering discipline, mechatronics is likely not far behind. However, for this to be the case, the main focus of robotics research must shift away from esoteric, pedagogical pursuits and toward the solution of practical problems. The highly competitive nature of the international market place, as well as the concerns of national security, demand it.

Several interesting areas for further research come to mind. The next level of system development should be the integration of a vision system. This capability, combined with cross-coordinated control, should provide more than enough flexibility to accomodate the largest, realistic task geometry uncertainties likely to be encountered in an industrial environment.

Cooperating robots, working in parallel, represent another interesting extension with a very real potential for near term industrial implementation. This configuration could especially benefit from the flexibility and convenience of screw theory in formulating appropriate kinestatic constraints. Indeed, without screw theory, constraint formulation would surely prove unwieldy.

Another extension of this work would be the use of the alternative control architecture suggested by Sweet and Good [1984]. That architecture, which integrated

wrench error feedback at the motion processor, rather than the axis processors, offers a number of interesting possibilities. Exotic control strategies facilitating high speeds and wrench decomposition [Griffis, 1988] could be more readily implemented. While this implies creating a complete robot controller from scratch, rather than augmenting an existing commercial controller, the expense and effort required to do this is declining rapidly. Recent advances in tiny, power-switching, integrated circuits, as well as in microcontroller and microprocessor technology, puts this capability within the reach of more and more researchers every day. This revolution in digital motion control technology [Horn, 1987], particularly in the area of plug-in board motion controllers, has virtually eliminated any excuse for not experimentally verifying any serious research in robot control.

One other area worth examining is the area of mechanical compliance. Stability problems associated with the use of electronic compliance alone have not been overcome [Brownell, 1988], nor are they likely to be in the near future. The main drawback of mechanical compliance, as realized by the use of elastomeric shear pads, is its lack of flexibility in modifying end-effector compliance characteristics. The shear pads of an RCC device are easily replaced, and they are marketed in a range of stiffness values. However, they cannot be replaced, and hence the the stiffness properties of the RCC device cannot be altered, while a task is in progress.

One possibility might be the use of a pneumatic chamber within the shear pad. By altering the air pressure, the stiffness characteristics could be varied, without the need to stop and replace the unit, itself. There may well be other possibilities, and they should be explored.

Clearly, the field of robotics is a very dynamic one. This is no doubt due, at least in part, to the fact that it is a relatively new technology. Indeed, many of the original pioneers from the 1950's are still alive today, and their influence can still be felt. However, as this technology matures, one must not lose sight of the fact that robotics is an applied technology. As such, the ultimate value of any robotics research must be measured in terms of its contribution to practical applications.

REFERENCES

Alberts, T.E., 1986, "Augmenting the Control of a Flexible Manipulator with Passive Mechanical Damping," Ph.D. Dissertation, Georgia Institute of Technology.

An, C.H., and Hollerbach, J.M., 1987a, "Dynamic Stability Issues in Force Control of Manipulators," Proc. IEEE International Conf. on Robotics and Automation, Raleigh, pp. 890-896.

An. C.H., and Hollerbach, J.M., 1987b, "Kinematic Stability Issues in Force Control of Manipulators," Proc. IEEE International Conf. on Robotics and Automation, Raleigh, pp. 897-903.

An, C.H., Atkenson, C.G., and Hollerbach, J.M., 1988, Model-Based Control of a Robot Manipulator, MIT Press, Cambridge, MA.

Asada, H., and Lim, S.K., 1985, "Design of Joint Torque Sensors and Torque Feedback Control for Direct-Drive Arms," Robotics and Manufacturing Automation, ASME PED-Vol. 15, New York.

Asada, H., and Slotine, J., 1986, Robot Analysis and Control, John Wiley and Sons, Inc., New York.

Asada, H., and Youcef-Toumi, K., 1984, "Analysis and Design of a Direct-Drive Arm with a Five-Bar-Link Parallel Drive Mechanism," ASME J. of Dynamic Systems Measurement and Control, 106, pp. 225-230.

Asada, H., and Youcef-Toumi, K., 1987, Direct-Drive Robots: Theory and Practice. MIT Press, Cambridge, MA.

Åström, K., and Wittenmark, B., 1984, Computer Controlled Systems - Theory and Design, Prentice-Hall, Englewood Cliffs, NJ.

Ball, R.S., 1900, A Treatise on the Theory of Screws, Cambridge University Press. London.

Bollinger, J.G., and Duffie, N.A., 1988, Computer Control of Machines and Processes. Addison-Wesley, Reading, MA.

Brady, M., Hollerbach, J.M., Johnson, T.L., Lozano-Perez, T., Mason, M.T., 1982, Robot Motion: Planning and Control, MIT Press, Cambridge, MA.

Brand, L., 1948, Vector and Tensor Analysis, John Wiley and Sons, Inc., New York.

Brownell, T.A., 1988, "Instabilities of Active Compliance," Proc. Robots 12 and Vision '88 Conf., Detroit, pp. 15.71-15.81.

Chesmond, C.J., 1982, Control System Technology, Butler and Tanner, Ltd, London.

Craig, J.J., 1986, Introduction to Robotics, Mechanics & Control, Addison-Wesley, Reading, MA.

Craig, J.J., 1988, Adaptive Control of Mechanical Manipulators, Addison-Wesley, Reading, MA.

Craig, J.J., and Raibert, M.H., 1979, "A Systematic Method of Hybrid Position/Force Control of a Manipulator," Proc. IEEE 3rd International Computer Software Applications Conf., Chicago, pp. 446-451.

De Schutter, J., 1986, "Compliant Robot Motion: Task Formulation and Control," Ph.D. Dissertation, Katholieke Universiteit Leuven.

De Schutter, J., and Van Brussel, H., 1988, "Compliant Robot Motion I: A Formalism for Specifying Compliant Motion Tasks, Compliant Robot Motion II: A Control Approach Based on External Control Loops," International J. of Robotics Research, Vol. 7, No. 4, MIT Press, Cambridge, MA.

Duffy, J., 1980, Analysis of Mechanisms and Robot Manipulators, John Wiley and Sons, Inc., New York.

Eppinger, S.D. and Seering, W.P., 1986, "On Dynamic Models of Robot Force Control," Proc. IEEE International Conf. on Robotics and Automation, San Francisco, pp. 29-34.

Eppinger, S.D. and Seering, W.P., 1987, "Understanding Bandwidth Limitations in Robot Force Control," Proc. IEEE International Conf. on Robotics and Automation, Raleigh, pp. 904-909.

Fourney, W.L., 1988, "Mechanical Engineering - In the Pole Position for Change," Engineering Horizons, Careers, Opportunities and Guidance, Inc., Boston, pp. 38-39.

Franklin, G.F., and Powell, J.D., 1980, Digital Control of Dynamic Systems, Addison-Wesley, Reading, MA.

Fu, K.S., Gonzalez, R.C., and Lee, C.S.G., 1987, Robotics - Control, Sensing, Vision, and Intelligence, McGraw-Hill, New York.

Gibson, C., and Hunt, K.H., 1988, "Geometry of Screw Systems," Unpublished.

Gillespie, L.K., 1987, Robotic Deburring Handbook, SME, Dearborn, MI.

Goertz, J.J., 1963, "Manipulators Used for Handling Robot Materials," Human Factors in Technology, Chapter 17, edited by E.M. Bennett, McGraw-Hill, New York.

Good, M.C., Sweet, L.M. and Strobel, K.L., 1985, "Dynamic Models for Control System Design of Integrated Robot and Drive Systems," ASME J. of Dynamic Systems, Measurement and Control, 107, pp. 53-59.

Graf, T.L., 1988, "Deburring and Finishing Applications Using Robots and Computerized Automation, 3M Corporation, St. Paul, MN.

Griffis, M., 1988, "Kinetic Considerations in the Hybrid Control of Robotic Manipulators," M.S. Thesis, University of Florida.

Hartenberg, R.S., and Denavit, J., 1964, Kinematic Synthesis of Linkages, McGraw-Hill, New York.

Haefner, J.B., Houpt, P.K., Baker, T.E., and Dausch, M.E., 1986, "Real Time Robot Position/Force Control for Deburring," Robotics: Theory and Applications, ASME, New York, pp. 73-78.

Hogan, N., 1985, "Impedance Control: An Approach to Manipulation: Part I-Theory, Part II-Implementation, Part III-Applications," ASME J. of Dynamic Systems, Measurement and Control, 107, pp. 1-24.

Holman, J.P., 1984, Experimental Methods for Engineers, McGraw-Hill Book Company, New York.

Horn, D., 1987, "Motion Control - A Revolution in Miniature," Mechanical Engineering, ASME, New York, 109, pp. 44-47.

Hunt, K.H., 1978, Kinematic Geometry of Mechanisms, Clarendon Press, Oxford.

Hunt, K.H., 1984, "Mechanisms-Research Directions in Kinematics and Geometry," ASME J. of Mechanisms, Transmissions, and Automation in Design, 106, pp. 262-263.

Inoue, H., 1971, "Computer Controlled Bilateral Manipulator," Bulletin of the Japan Society of Mechanical Engineers, Vol. 14, March.

Johnson, M.R., 1987, Fundamentals of Microprocessor-Based Design - A Tutorial, presented at the 33rd International Instrumentation Symposium, Las Vegas.

Kazerooni, H., Houpt, P.K., and Sheridan, T.B., 1986, "The Fundamental Concepts of Robust Compliant Motion for Robot Manipulators," Proc. IEEE International Conf. on Robotics and Automation, San Francisco, pp. 418-427.

Khatib, O., 1983, "Dynamic Control of Manipulators in Operational Space," Proc. 6th IFToMM Congress on the Theory of Machines and Mechanisms, New Delhi, pp. 1128-1131.

Khatib, O., 1987, "A Unified Approach for Motion and Force Control of Robot Manipulators: The Operational Space Formulation," IEEE J. of Robotics and Automation, RA-3, pp. 43-53.

Khatib, O., and Burdick, J., 1986, "Motion and Force Control of Robot Manipulators," Proc. IEEE International Conf. on Robotics and Automation, San Francisco, pp. 1381-1386.

Klein, F., 1908, Elementary Mathematics from an Advanced Syandpoint - Geometry, reprinted, Dover Publishing Co., New York, 1939.

Kline, S.J. and McClintock, F.A., 1953, "Describing Uncertainties in Single-Sample Experiments," Mechanical Engineering, pp. 3-7.

Koren, Y., 1985, Robotics for Engineers, McGraw-Hill, New York.

Koren, Y., and Ulsoy, A.G., 1982, "Control of DC Servo-Motor Driven Robots," reprinted by SME from Proc. Robots VI, Detroit.

Lawrence, P.D., and Mauch, K., 1987, Real-Time Microcomputer System Design: An Introduction, Mcgraw-Hill, New York.

Leu, M.C., Dukovski, V., and Wang, K.K., 1985, "An Analytical and Experimental Study of the Stiffness of Robot Manipulators with Parallel Mechanisms," Robotics and Manufacturing Automation, ASME PED-VOL. 15, New York.

Lipkin, H. and Duffy, J., 1982, "Analysis of Industrial Robots via the Theory of Screws," Proc. of the 12th International Symposium on Industrial Robots, Paris, pp. 359-370.

Lipkin, H., and Duffy, J., 1984, "On the Geometry of Orthogonal and Reciprocal Screws," Proc. 5th CISM-IFToMM Symposium, University of Warsaw, pp. 47-55.

Lipkin, H. and Duffy, J., 1985a, "The Elliptic Polarity of Screws," Trans. ASME J. Mechanisms, Transmissions, and Automation in Design, 107, pp. 377-387.

Lipkin, H., and Duffy, J., 1885b, "A Vector Analysis of Robot Manipulators," in Recent Advances in Robotics, edited by G. Beni and S. Hackwood, John Wiley and Sons, Inc., New York, pp. 175-241.

Lipkin, H. and Duffy, J., 1986, "Hybrid Twist and Wrench Control of Robotic Manipulators," ASME Paper No. 86-DET-74, ASME Mechanisms Conference, Columbus, OH.

Lipkin, H. and Duffy, J., 1987, "Invariant Kinestatic Filtering," Proc. of the 7th World Congress on the Theory of Machines and Mechanisms, Sevilla, Spain, pp. 297-300.

Lipkin, H , and Duffy, J., 1988, "Hybrid Twist and Wrench Control of a Robotic Manipulator," Trans. ASME J. Mechanisms, Transmissions, and Automation in Design, 110, pp. 138-144.

Lord Corporation, 1986, "Installation and Operations Manual for F/T Series Force/Torque Sensing Systems," Industrial Automation Division, Cary, NC.

Luh, J.Y.S., Walker, M.W., and Paul, R.P., 1980, "Resolved Acceleration Control of Mechanical Manipulators," IEEE Transactions on Automatic Control, Vol. AC-25, No. 3, June.

Maples, J.A., and Becker, J.J., 1986, "Experiments in Force Control of Robotic Manipulators," Proc. IEEE International Conf. on Robotics and Automation, San Francisco, pp. 695-702.

Mason, M.T., 1976, "Compliance and Force Control for Computer-Controlled Manipulators," M.S. Thesis, MIT.

Mason, M.T., 1979, "Compliance and Force Control for Computer-Controlled Manipulators," MIT Artificial Intelligence Lab., TR 515, M.S. Thesis with revisions.

Mason, M.T., 1981, "Compliance and Force Control for Computer-Controlled Manipulators," IEEE Trans. on Systems, Man, and Cybernetics, Vol., Smc-11, No. 6, June, pp. 418-432.

Mason, M.T., 1983, "Compliant Motion," in Robot Motion: Planning and Control, eds. Brady, et al., MIT Press, Cambridge, MA.

Mayday, C.J., 1987, Computer-Aided Design of Feedback Control Systems, ISA. Research Triangle Park, NC.

Merritt, H.E., 1967, Hydraulic Control Systems, John Wiley & Sons, Inc., New York.

Ogata, K., 1970, Modern Control Engineering, Prentice-Hall, Inc.. Englewood Cliffs, NJ.

Ogata, K., 1987, Discrete-Time Control Systems, Prentice-Hall, Inc., Englewood Cliffs. NJ.

Ohwovoriole, M.S., 1980, "An Extension to Screw Theory and its Application to the Automation of Industrial Assemblies," Ph.D. Dissertation, Stanford University.

Ohwovoriole, M.S., and Roth, B., 1981, "An Extension of Screw Theory," ASME J. of Mechanical Design, 103, pp. 725-735.

Paul, R.P., 1981, Robot Manipulators - Mathematics, Programming, and Control, MIT Press, Cambridge, MA.

Paul, R.P., 1987, "Problems and Research Issues Associated with the Hybrid Control of Force and Displacements," Proc. IEEE International Conf. on Robotics and Automation, Raleigh, pp. 1966-1971.

Paul, R.P., and Shimano, B., 1976, "Compliance and Control," Proc. Joint Automatic Control Conf., Purdue University.

Raibert, M.H. and Craig, J.J., 1981, "Hybrid Position/Force Control of Manipulators," ASME J. of Dynamic Systems, Measurement and Control, 105, pp. 126-133. (Also in Brady, et al. [1982], pp. 419-438.)

Rivin, E.I., 1988, Mechanical Design of Robots, McGraw-Hill, Inc., New York.

Roberts, R.K., 1984, "The Compliance of End Effector Force Sensors for Robot Manipulator Control," Ph.D. Dissertation, Purdue University.

Rosenblum, A., 1975, "DC Motors, Speed Controls, Servo Systems - An Engineering Handbook," Electro-Craft Corporation, Hopkins, MN.

Roth, B., 1984, "Screws, Motors, and Wrenches that Cannot Be Bought in a Hardware Store," Robotics Research: The First International Symposium, MIT Press, Cambridge, MA, pp. 679-693.

Ruocco, S.R., 1987, Robot Sensors and Transducers, Halstead Press, New York.

Salisbury, J.K., 1980, "Active Stiffness Control of a Manipulator in Cartesian Coordinates," Proc. of the 19th Conf. on Decision and Control, Vol. 1, Albuquerque. (Also in Mason, M.T. and Salisbury, J.K., [1985], pp. 95-108.)

Seltzer, D.S., 1986, "Compliant Robot Wrist Sensing For Precision Assembly," Robotics: Theory and Applications, ASME, New York, pp. 161-168.

Shahinpoor, M., 1987, A Robot Engineering Textbook, Harper and Row, New York.

Shin, K.G., and Lee, C.P., 1985, "Compliant Control of Robotic Manipulators with Resolved Acceleration," Proc. 24th IEEE Conf. Decision and Control, Ft. Lauderdale, pp. 350-357.

Shimano, B.E., 1978, "The Kinematic Design and Force Control of Computer Controlled Manipulators," Stanford Artificial Intelligence Lab Memo AIM-313, March.

Snyder, W.E., 1985, Industrial Robots: Computer Interfacing and Control, Prentice-Hall, Englewood Cliffs, NJ.

Stepien, T.M., Sweet, L.M., Good, M.C., and Tomizuka M., 1987, "Control of Tool/Workpiece Contact Force with Application to Robotic Deburring," IEEE J. of Robotics and Automation, Vol. RA-3, No. 1, February.

Sugimoto, K., and Matsumoto, Y., 1984, "Kinematic Analysis of Manipulators by Means of the Projective Transformation of Screw Coordinates," Robotics Research, The First International Symposium, MIT Press, Cambridge, MA., pp. 695-705.

Sweet, L.M. and Good, M.C., 1984, "Re-Definition of the Robot Motion Control Problem: Effects of Plant Dynamics, Drive System Constraints, and User Requirements," Proc. of 23rd Conf. on Decision and Control, Las Vegas, pp. 724-732.

Tarn, T.J., Bejczy, A.K., and Yun, X., 1988, "Robot Arm Force Control Through System Linearization by Nonlinear Feedback," Proc. IEEE International Conf. on Robotics and Automation, Philadelphia, pp. 1618-1625.

Tlusty, J., 1972, "Grinding Machine Ability to Reduce Workpiece Form Error," New Developments in Grinding - Proc. of the International Grinding Conference, April 18-20.

Waldron, K.J., 1969, "The Mobility of Linkages," Ph.D. Dissertation, Stanford University.

Wedel, D.L. and Saridis, G.N., 1988, "An Experiment in Hybrid Position/Force Control of a Six DOF Revolute Manipulator," Proc. IEEE International Conf. on Robotics and Automation, Philadelphia, pp. 1638-1642.

West, H., and Asada, H., 1985, "A Method for the Design of Hybrid Position/Force Controllers for Manipulators Constrained by Contact with the Environment," Proc. IEEE International Conf. on Robotics and Automation, St. Louis, pp. 251-259.

Whitney, D.E., 1977, "Force Feedback Control of Manipulator Fine Motions," ASME J. of Dynamic Systems, Measurement, and Control, 99, pp. 91-97.

Whitney, D.E., 1982, "Quasi-Static Assembly of Compliantly Supported Rigid Parts," ASME J. of Dynamic Systems, Measurement and Control, 104, pp. 65-77.

Whitney, D.E., 1987, "Historical Perspective on the State of the Art in Robot Force Control," The International J. of Robotics Research, Vol. 6, No. 1, Spring, pp. 3-14.

Whitney, D.E., and Nevins, J.L., 1978, "What is the Remote Center Compliance (RCC) and What Can It Do?," Internal Report P-728, The Charles Stark Draper Lab., Cambridge, MA.

Whitney, D.E., and Rourke, J.M., 1986, "Mechanical Behaviour and Design Equations for Elastomer Shear Pad Remote Center Compliances," ASME J. of Dynamic Systems, Measurement, and Control, 108, pp. 223-232.

Youcef-Toumi, K, and Nagano, H., 1987, "Design and Control of Drive Systems Using Low-Reduction Gears for Force Control," Modeling and Control of Robotic Manipulators and Manufacturing Processes, ASME DSC-Vol. 6, New York.

Youcef-Toumi, K., and Ro, P.I., 1986, "A Dual-Drive Concept for Enhancing the Micro-Manipulation of Direct-Drive Arms," Robotics: Theory and Applications, ASME DSC-Vol. 3, New York.

Yabuta, T., Chona, A.J. and Beni, G., 1988, "On the Asymptotic Stability of the Hybrid Position/Force Control Scheme for Robot Manipulators," Proc. IEEE International Conf. on Robotics and Automation, Philadelphia, pp. 338-343.

Zhang, H. and Paul R.P., 1985, "Hybrid Control of Robot Manipulators," Proc. IEEE International Conf. on Robotics and Automation, St. Louis, pp. 602-607.

BIOGRAPHICAL SKETCH

Mark L. Swinson was born [redacted] After graduating from [redacted] High School in 1970, he began his undergraduate studies at the United States Military Academy (USMA), West Point, New York. He received his Bachelor of Science in Engineering degree in 1974, where he was designated a Distinguished Graduate for having finished in the top 5% of his class by General Order of Merit. From 1974 until 1980, he served on active duty as an officer of the United States Army. In 1980, he began his graduate studies at the University of Wisconsin-Madison, receiving his Master of Science in Mechanical Engineering degree in 1982. He then joined the faculty of the Department of Mechanics, USMA, where he served as an assistant professor and course director for both the fluid dynamics and thermodynamics core courses. Having completed the requirements for education, experience and by successful examination, he was licensed in Virginia as a Registered Professional Engineer in 1985. That same year, while still on active duty, he continued his graduate studies at the University of Florida.